



TUGAS AKHIR - SS141501

**PERBANDINGAN ALGORITMA *FP-GROWTH*, *APRIORI*,
DAN *SQUEEZER* PADA ANALISIS PERILAKU KONSUMEN
DI MINIMARKET *K1MART ITS***

**SITI QOMARIYAH
NRP 1313100 026**

**Dosen Pembimbing
Dr. Kartika Fithriasari, M.Si**

**PROGRAM STUDI S1
JURUSAN STATISTIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA 2017**



TUGAS AKHIR - SS 141501

**PERBANDINGAN ALGORITMA *FP-GROWTH*,
APRIORI, DAN *SQUEEZER* PADA ANALISIS
PERILAKU KONSUMEN DI MINIMARKET *K1MART*
ITS**

**SITI QOMARIYAH
NRP 1313100 026**

**Dosen Pembimbing
Dr. Kartika Fithriasari, M.Si**

**PROGRAM STUDI S1
JURUSAN STATISTIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA 2017**



FINAL PROJECT - SS 141501

***COMPARISON OF FP-GROWTH, APRIORI,
AND SQUEEZER ALGORITHM ON ANALYSIS OF
CONSUMER BUYING PATTERN IN MINIMARKET
K1MART ITS***

**SITI QOMARIYAH
NRP 1313100 026**

**Supervisor
Dr. Kartika Fithriasari, M.Si**

**UNDERGRADUATE PROGRAMME
DEPARTMENT OF STATISTICS
FACULTY OF MATHEMATICS AND NATURAL SCIENCES
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA 2017**

LEMBAR PENGESAHAN

**PERBANDINGAN ALGORITMA FP-GROWTH,
APRIORI, DAN SQUEEZER PADA ANALISIS
PERILAKU KONSUMEN DI MINIMARKET
K1MART ITS**

TUGAS AKHIR

**Diajukan untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Sains
pada**

**Program Studi S-1 Jurusan Statistika
Fakultas Matematika dan Ilmu Pengetahuan Alam
Institut Teknologi Sepuluh Nopember**

**Oleh :
SITI QOMARIYAH
NRP. 1313 100 026**

**Disetujui oleh Pembimbing Tugas Akhir:
Dr. Kartika Fithriasari, M.Si.
NIP. 19691212 199303 2 002**

**Mengetahui
Ketua Jurusan Statistika FMIPA-ITS**

Dr. Suhartono

NIP. 19710929 199512 1 001

Surabaya, Januari 2017



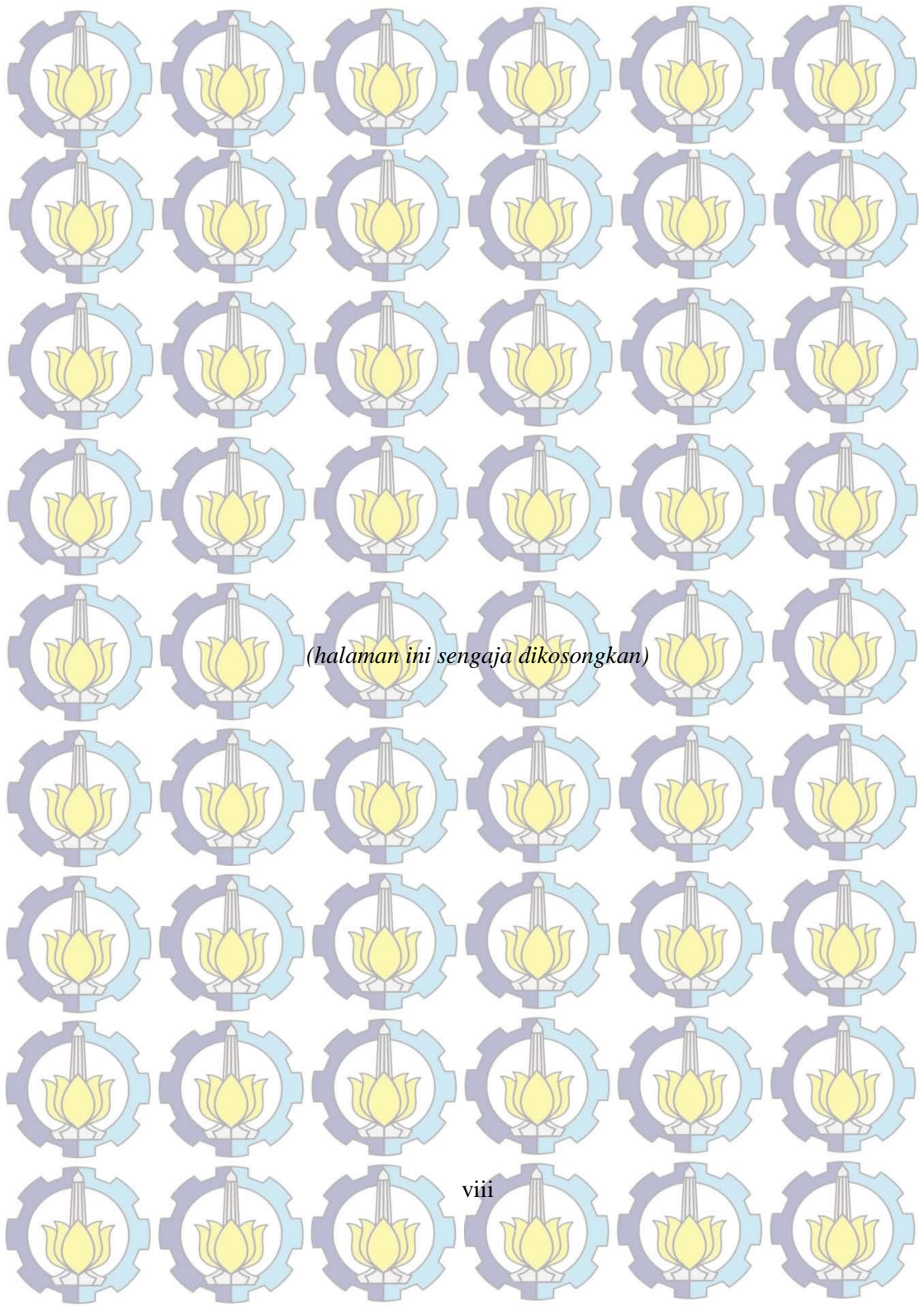
Perbandingan Algoritma *FP-Growth*, *Apriori*, dan *Squeezer* pada Analisis Perilaku Konsumen di Minimarket *K1mart* ITS

Nama Mahasiswa : Siti Qomariyah
NRP : 1313 100 026
Jurusan : Statistika FMIPA - ITS
Dosen Pembimbing : Dr. Kartika Fithriasari, M.Si

Abstrak

Di era modern ini, terdapat 36.000 lebih gerai ritel di Indonesia. Hal ini menyebabkan persaingan dalam merebut pangsa pasar menjadi semakin ketat. Salah satu usaha ritel yang ada di kawasan ITS adalah minimarket *K1mart*. Hal yang dibutuhkan untuk bisa membuat strategi pemasaran yang tepat adalah dengan mengetahui pola pembelian konsumen. Manfaat yang didapatkan setelah mengetahui pola pembelian konsumen adalah dapat menyesuaikan promosi, pengaturan toko, membuat iklan yang sesuai, membuat layout untuk situs *e-commerce*, mengontrol persediaan barang. Terdapat tiga algoritma yang bisa digunakan untuk melakukan analisis perilaku konsumen. Algoritma tersebut antara lain *FP-Growth*, *Apriori*, dan *Squeezer*. Data yang digunakan untuk melakukan analisis adalah data transaksi mulai 1 Maret 2016 sampai 28 Maret 2016 yaitu sejumlah 9871 transaksi. Ukuran yang digunakan untuk mengetahui kebaikan *association rule* yang didapatkan adalah *support*, *confidence*, dan *lift*. Jumlah pola pembelian konsumen yang dihasilkan pada ketiga algoritma tidak sama. Algoritma terbaik yang didapatkan adalah *FP-Growth* sebab mampu menghasilkan *association rule* dengan nilai *support*, *confidence*, dan *lift* terbesar. Sedangkan algoritma yang memiliki waktu proses paling cepat adalah algoritma *Apriori*.

Kata Kunci : *Apriori*, *Association Rule*, *FP-Growth*, *Squeezer*, *Support*



(halaman ini sengaja dikosongkan)

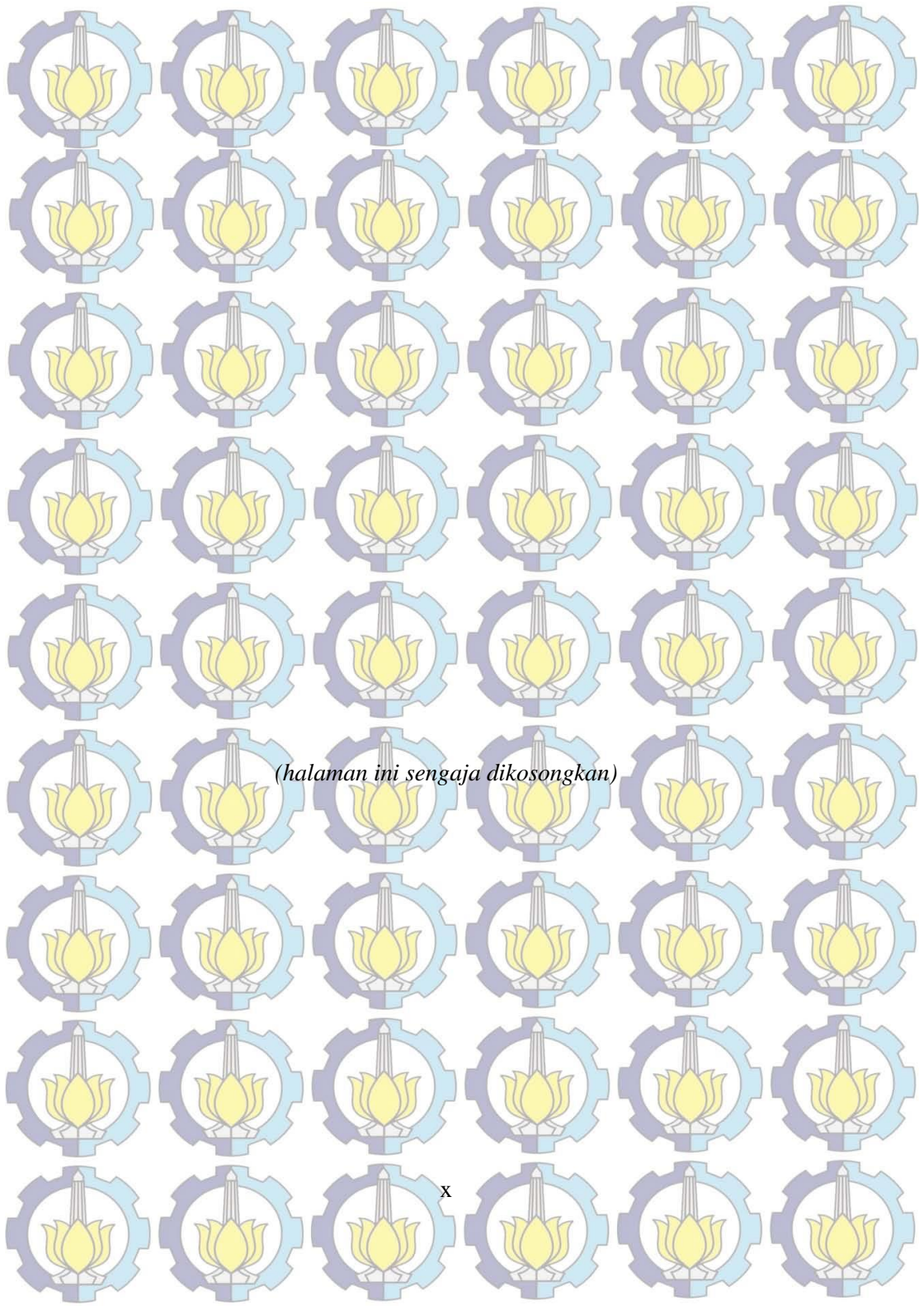
Comparison of FP-Growth, Apriori, and Squeezer Algorithm on Analysis Of Analysis Of Consumer Buying Pattern In Minimarket K1mart ITS

Name : Siti Qomariyah
NRP : 1313 100 026
Department : Statistics FMIPA - ITS
Supervisor : Dr. Kartika Fithriasari, M.Si

Abstract

In this modern era, there is more than 36.000 retail in Indonesia. This condition lead the competition to grab consumer become more competitif. One of retail in the area of ITS is K1mart minimarket. To make the right marketing strategi, K1mart need to know consumer buying behavior. Benefits gained after learning the purchasing patterns of consumers are able to customize promotions, store layout settings, create layout for e-commerce sites, and control inventory. There is 3 algorithm that can be use to analyze consumer buying pattern. That algorithm arex FP-Growth, Apriori, and Squeezer. The data used for the analysis is transaction data from 1 March 2016 until 28 March 2016. That is 9871 transaction data. The measure used to determine the association rule is support, confidence and lift. Number of consumer buying pattern generated by FP-Growth algorithm, apriori algorithm and squeezer algorithm are not same. The best algorithm is FP-Growth, because FP-Growth has the highest support, confidence and lift. From that three algorithm, algorithm that have the minimum running time is Apriori Algorithm.

Keywords: Apriori, Association Rule, FP-Growth, Squeezer, Support




(halaman ini sengaja dikosongkan)

KATA PENGANTAR

Puji syukur kehadiran Allah SWT, Tuhan Yang Maha Esa. Berkat rahmat dan ridho-Nya penulis dapat menyelesaikan laporan Tugas Akhir yang berjudul **“Perbandingan Algoritma *FP-Growth*, *Apriori* dan *Squeezer* pada Analisis Perilaku Konsumen di Minimarket *KI*mart ITS”** dengan lancar.

Keberhasilan penyusunan Tugas Akhir ini tidak lepas dari banyaknya bantuan dan dukungan yang diberikan dari berbagai pihak. Oleh karena itu, pada kesempatan ini penulis mengucapkan terima kasih kepada:

1. Bapak Dr. Suhartono selaku Ketua Jurusan Statistika dan Bapak Dr. Sutikno, M.Si selaku Koordinator Program Studi S1 yang telah memberikan fasilitas untuk kelancaran penyelesaian Tugas Akhir.
2. Ibu Dr. Kartika Fithriasari, M.Si selaku dosen pembimbing yang telah dengan sabar memberikan bimbingan, saran, dan dukungan selama penyusunan Tugas Akhir.
3. Bapak Prof. Drs. Nur Iriawan, M.Ikom, PhD dan Ibu Dr. Santi Wulan Purnami selaku dosen penguji yang telah memberikan banyak bantuan dan saran untuk kesempurnaan Tugas Akhir ini.
4. Bapak Drs. Agus Suharsono, MS selaku dosen wali yang telah memberikan nasehat dan semangat.
5. Seluruh dosen Statistika ITS yang telah memberikan ilmu dan pengetahuan yang tak ternilai harganya, serta segenap karyawan Jurusan Statistika ITS.
6. Ibu Sulasmi dan Bapak Rifa'i, orangtua penulis yang selalu memberikan dukungan, kasih sayang dan doa yang tidak pernah putus, serta kakak penulis yaitu Abdul Rozaq yang senantiasa menjadi penyemangat bagi penulis dalam menyelesaikan Tugas Akhir ini.

- 
7. Teman-teman dan semua pihak yang telah memberikan bantuan hingga penyusunan laporan Tugas Akhir ini dapat terselesaikan yang tidak dapat penulis sebutkan satu persatu. Penulis berharap hasil Tugas Akhir ini dapat bermanfaat bagi kita semua. Semoga kebaikan dan bantuan yang telah diberikan kepada penulis dibalas dengan kebaikan yang lebih besar lagi oleh Tuhan Yang Maha Esa. Aamiin.

Surabaya, Januari 2017

Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
COVER PAGE	iii
LEMBAR PENGESAHAN	v
ABSTRAK	vii
ABSTRACT	ix
KATA PENGANTAR	xi
DAFTAR ISI	xiii
DAFTAR GAMBAR	xv
DAFTAR TABEL	xvii
DAFTAR LAMPIRAN	xix
BAB I PENDAHULUAN	
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah	4
1.3 Tujuan Penelitian.....	5
1.4 Manfaat Penelitian.....	5
1.5 Batasan Penelitian	5
BAB II TINJAUAN PUSTAKA	
2.1 <i>Preprocessing</i>	7
2.2 <i>Assosiation Rule</i>	8
2.3 <i>Market Basket Analysis</i>	9
2.4 <i>FP-Growth</i>	11
2.5 <i>Apriori</i>	22
2.6 <i>Squeezer</i>	25
2.7 Interpretasi Hasil <i>Association Rule</i>	28
BAB III METODOLOGI PENELITIAN	
3.1 Sumber Data	29
3.2 Variabel Penelitian	29
3.3 Langkah Analisis.....	29
3.4 Diagram Alir.....	30

BAB IV ANALISIS DAN PEMBAHASAN

4.1	<i>Preprocessing</i>	33
4.1.1	Membaca Data dari Notepad	33
4.1.2	Menghapus Kantong Plastik	35
4.1.3	Membuat Kode Barang	35
4.1.4	Mengganti Nama Barang dengan Kode Barang	38
4.1.4	Membuat Struktur Data Biner	38
4.2	Statistika Deskriptif Persebaran Pembelian Konsumen...	40
4.2.1	Data Barang Detail	40
4.2.2	Data Barang Umum	43
4.3	Pola Perilaku Konsumen dengan Algoritma <i>FP-Growth</i>	46
4.3.1	Data Barang Detail	47
4.3.2	Data Barang Umum	51
4.4	Pola Perilaku Konsumen dengan Algoritma Apriori	57
4.4.1	Data Barang Detail	57
4.4.2	Data Barang Umum	61
4.5	Pola Perilaku Konsumen dengan Algoritma <i>Squeezer</i>	66
4.5.1	Data Barang Detail	67
4.5.2	Data Barang Umum	67
4.6	Perbandingan Algoritma <i>FP-Growth</i> , Apriori dan <i>Squeezer</i>	68
4.7	Letak Barang yang Sering Dibeli Konsumen di Minimarket <i>Klmart</i>	70
4.8	Pola Perilaku Konsumen yang Dihasilkan pada Item dengan <i>Support</i> Tertinggi	72

BAB V KESIMPULAN DAN SARAN

5.1	Kesimpulan	75
5.2	Saran	76

DAFTAR PUSTAKA	79
-----------------------------	----

LAMPIRAN	83
-----------------------	----

BIODATA PENULIS	103
------------------------------	-----

DAFTAR GAMBAR

	Halaman
Gambar 2.1	<i>FP-tree</i> Setelah Pemindaian TID 113
Gambar 2.2	<i>FP-tree</i> Setelah Pemindaian TID 214
Gambar 2.3	<i>FP-tree</i> Setelah Pemindaian TID 315
Gambar 2.4	<i>FP-tree</i> Setelah Pemindaian TID 415
Gambar 2.5	<i>FP-tree</i> Setelah Pemindaian TID 516
Gambar 2.6	Pembentukan <i>FP-tree</i>17
Gambar 2.7	<i>Conditional Pattern Base</i> untuk <i>Suffix</i> p19
Gambar 2.8	<i>Conditional Pattern Base</i> untuk <i>Suffix</i> m20
Gambar 2.9	<i>Conditional Pattern Base</i> untuk <i>Suffix</i> b20
Gambar 2.10	<i>Conditional Pattern Base</i> untuk <i>Suffix</i> a21
Gambar 2.11	<i>Conditional Pattern Base</i> untuk <i>Suffix</i> c21
Gambar 3.1	Diagram Alir31
Gambar 4.1	<i>Flowchart</i> Membaca File Notepad33
Gambar 4.2	<i>Flowchart</i> Membuat Daftar Barang Umum36
Gambar 4.3	<i>Flowchart</i> Membuat Struktur Data Biner38
Gambar 4.4	Persebaran Jumlah Item yang Dibeli pada Data Barang Detail40
Gambar 4.5	Diagram Batang Pembelian pada Data Barang Detail41
Gambar 4.6	Persebaran <i>Support Count</i> pada Data Barang Detail43
Gambar 4.7	Persebaran Jumlah Item pada Data Barang Umum44
Gambar 4.8	Diagram Batang Jumlah Item 1 Sampai dengan 844
Gambar 4.9	Persebaran Jumlah Item yang Dibeli pada Data Barang Umum46
Gambar 4.10	Letak Roti pada Minimarket <i>K1mart</i>70



Gambar 4.11 Letak Soba Mi pada Minimarket *Klmart*71

Gambar 4.12 Letak Susu Ultra pada Minimarket *Klmart*71

Gambar 4.13 Letak Club Air Mineral pada Minimarket *Klmart*71

DAFTAR TABEL

	Halaman
Tabel 2.1 <i>Binary Database</i>	10
Tabel 2.2 <i>Database Transaksi</i>	11
Tabel 2.3 <i>Database Transaksi Sebelum Reduksi Dimensi</i>	12
Tabel 2.4 <i>Database Transaksi Setelah Reduksi Dimensi</i>	13
Tabel 2.5 <i>Conditional Pattern-base, Conditional FP-tree dan Frequent Itemset yang Terbentuk</i>	22
Tabel 2.6 <i>Database Transaksi</i>	23
Tabel 2.7 <i>Support Count 1-Itemset</i>	24
Tabel 2.8 <i>Support Count 2-Itemset</i>	24
Tabel 2.9 <i>Support Count 3-Itemset</i>	25
Tabel 2.10 <i>Transaksi Penjualan</i>	26
Tabel 2.11 <i>Kode Barang</i>	26
Tabel 4.1 <i>Statistika Deskriptif Jumlah Item Data Barang Detail dan Barang Umum</i>	40
Tabel 4.2 <i>Statistika Deskriptif Support Count Data Barang Detail dan Barang Umum</i>	42
Tabel 4.3 <i>1 Itemset dengan Support Tertinggi pada Data Barang Detail Algoritma FP-Growth</i>	46
Tabel 4.4 <i>2 Itemset dengan Support Tertinggi pada Data Barang Detail Algoritma FP-Growth</i>	47
Tabel 4.5 <i>2 Rule pada Data Barang Detail Algoritma FP-Growth</i>	48
Tabel 4.6 <i>1 Itemset dengan Support Tertinggi pada Data Barang Umum Algoritma FP-Growth</i>	51
Tabel 4.7 <i>2 Itemset dengan Support Tertinggi pada Data Barang Umum Algoritma FP-Growth</i>	52
Tabel 4.8 <i>Rule pada Data Barang Umum Algoritma FPGrowth</i>	53

Tabel 4.9	1 <i>Itemset</i> dengan <i>Support</i> Tertinggi pada Data Barang Detail Algoritma Apriori	57
Tabel 4.10	2 <i>Itemset</i> dengan <i>Support</i> Tertinggi pada Data Barang Detail Algoritma Apriori	58
Tabel 4.11	<i>Rule</i> pada Data Barang Detail Algoritma Apriori.....	59
Tabel 4.12	1 <i>Itemset</i> dengan <i>Support</i> Tertinggi pada Data Barang Umum Algoritma Apriori.....	62
Tabel 4.13	2 <i>Itemset</i> dengan <i>Support</i> Tertinggi pada Data Barang Umum Algoritma Apriori.....	63
Tabel 4.14	<i>Rule</i> pada Data Barang Umum Algoritma Apriori.....	64
Tabel 4.15	Hasil <i>Cluster</i> Data Barang Detail Algoritma <i>Squeezer</i>	67
Tabel 4.16	Hasil <i>Cluster</i> pada Data Barang Umum Algoritma <i>Squeezer</i>	68
Tabel 4.17	Perbandingan Algoritma <i>FP-Growth</i> , <i>Apriori</i> , <i>Squeezer</i>	68

DAFTAR LAMPIRAN

	Halaman
Lampiran 1 Contoh Struk Transaksi	83
Lampiran 2 <i>Database</i> Pembelian Barang Detail	84
Lampiran 3 <i>Database</i> Pembelian Barang Umum	85
Lampiran 4 <i>Syntax</i> Membaca Notepad	86
Lampiran 5 <i>Syntax</i> Menulis Barang Berdasarkan Transaksi	87
Lampiran 6 <i>Syntax</i> Menulis Data dalam Bentuk <i>Market Basket</i>	87
Lampiran 7 <i>Syntax</i> Algoritma <i>Apriori</i>	88
Lampiran 8 <i>Syntax</i> Algoritma <i>Squeezer</i>	89
Lampiran 8 <i>Syntax</i> Algoritma <i>Squeezer</i> (Lanjutan)	90
Lampiran 9 <i>Output</i> Algoritma <i>FP-Growth</i> Data Barang Detail 1 <i>Itemset</i>	91
Lampiran 10 <i>Output</i> Algoritma <i>FP-Growth</i> Data Barang Detail 2 <i>Itemset</i>	92
Lampiran 11 <i>Output</i> Rule <i>FP-Growth</i> Barang Detail	92
Lampiran 12 <i>Output</i> Algoritma <i>FP-Growth</i> Data Barang Detail 1 <i>Itemset</i>	93
Lampiran 13 <i>Output</i> Algoritma <i>FP-Growth</i> Data Barang Detail 2 <i>Itemset</i>	94
Lampiran 14 <i>Output</i> Rule Data Barang Detail Algoritma <i>FP-Growth</i>	95
Lampiran 15 <i>Output</i> Top 23 <i>Frequent Itemset</i> Barang Detail Algoritma <i>Apriori</i> untuk 1 Item	96
Lampiran 16 <i>Output</i> <i>Frequent Itemset</i> Data Barang Detail Algoritma <i>Apriori</i> untuk 2 Item	97
Lampiran 17 <i>Output</i> Top 18 Rule Algoritma <i>Apriori</i> Data Barang Detail	98



Lampiran 18	<i>Output Top 23 Frequent Itemset Barang Umum</i>	Algoritma Apriori untuk 1 Item	99
--------------------	---	--------------------------------------	----

Lampiran 19	<i>Output Frequent Itemset Data Barang Umum</i>	Algoritma Apriori untuk 2 Item	100
--------------------	---	--------------------------------------	-----

Lampiran 20	<i>Output Top 18 Rule</i>	Algoritma Apriori Data Barang Umum	101
--------------------	---------------------------	--	-----

Lampiran 21	Surat Keterangan Pengambilan Data		102
--------------------	---	--	-----



BAB I PENDAHULUAN

1.1 Latar Belakang

Jumlah ritel yang tersebar di seluruh wilayah Indonesia telah mencapai lebih dari 36.000 gerai (Lussy, 2016). Banyaknya ritel menyebabkan semakin ketatnya persaingan dalam memperebutkan pangsa pasar. Salah satu bisnis ritel yang berada di kawasan ITS adalah minimarket *K1mart*. Minimarket *K1mart* baru diresmikan pada Jum'at 28 Agustus 2015. Menurut Muriyanto, minimarket *K1mart* merupakan bisnis baru yang dapat meningkatkan penghasilan ITS sebab keuntungan dalam jual beli dapat menambah pendapatan ITS. Olehkarena itu peneliti memilih minimarket *K1mart* ITS, agar manfaat dari penelitian yang dilakukan mampu dirasakan langsung oleh masyarakat ITS sebagai konsumen dari minimarket *K1mart* dan juga mampu berpartisipasi dalam meningkatkan pendapatan ITS. Sebagai minimarket yang tergolong baru, *K1mart* harus mampu membuat strategi pemasaran yang tepat agar mampu merebut pangsa pasar dan dapat terus berkembang. Hal yang dibutuhkan untuk membuat strategi pemasaran yang tepat adalah dengan mengetahui pola perilaku konsumennya. Kepuasan pelanggan merupakan salah satu kunci keberhasilan suatu usaha (Barskey, 1992). Tidak ada satupun bisnis atau organisasi yang dapat sukses tanpa membangun kepuasan dan kelayaitasan pelanggan (Timm, 2005). Olehkarena itu perlu diketahui tentang bagaimanakah pola pembelian pelanggan di *K1mart*.

Dalam bisnis ritel, mengetahui pola perilaku konsumen ini sangat penting. Beberapa manfaat yang didapatkan dengan mengetahui pola perilaku konsumen adalah dapat menyesuaikan promosi, pengaturan toko, pemberian pelayanan yang lebih baik, membuat iklan yang sesuai, membuat *layout* untuk situs *e-commerce*, mengontrol persediaan barang berdasarkan permintaan (Venkatachari, 2016), mengurangi produk dengan *brand* tertentu yang dirasa tidak menguntungkan (Kumar, 2003) dan

menghilangkan produk yang tidak sesuai dengan kebutuhan pelanggan (Larasati, Nasrun, & Ahmad, 2013).

Minimarket *K1mart* menggunakan komputer sebagai alat bantu untuk mencatat proses transaksinya. Data-data transaksi yang dilakukan konsumen setiap harinya akan tersimpan dalam memori komputer. Semakin lama data transaksi akan bertambah banyak setiap harinya. Akan sangat disayangkan jika data transaksi ini hanya dibiarkan saja. Padahal sebenarnya ada informasi yang dapat dimanfaatkan seperti informasi mengenai pola perilaku konsumen. Untuk mendapatkan informasi yang menarik atau mendapatkan informasi suatu pola tertentu pada *database* dapat menggunakan suatu alat yang dinamakan *data mining* (Han, 2012). Terdapat beberapa metode yang digunakan dalam *data mining* yaitu klasifikasi, prediksi, *association rule*. Dari beberapa metode dalam *data mining* tersebut, pada kasus ini peneliti menerapkan metode *association rule*. *Association rule* digunakan untuk mencari pola keterkaitan item-item yang sering muncul secara bersama-sama dalam suatu *database*. Dimana dalam kasus ritel, metode ini lebih dikenal sebagai *market basket analysis*. Data transaksi dari setiap konsumen dianggap sebagai suatu keranjang belanjaan. Isi dari keranjang belanjaan setiap konsumen akan berbeda-beda. Dari data transaksi yang sudah tertimbun banyak ini dapat dimanfaatkan untuk mencari pola perilaku konsumen seperti apa saja barang-barang yang sering dibeli konsumen secara bersama-sama.

Terdapat beberapa algoritma yang dapat digunakan dalam *market basket analysis* untuk mencari *frequent itemset*. Algoritma yang paling umum digunakan dalam *market basket analysis* adalah algoritma apriori. Apriori merupakan algoritma pertama yang paling efisien dalam menemukan pola asosiasi adalah apriori. Algoritma ini dikembangkan pada tahun 1994 oleh Agrawal dan Srikandi. Algoritma ini cukup efisien karena adanya pemangkasan pada kandidat *k-itemset* dengan *subset* yang berisi *k-1* item tidak memenuhi *minimum support*. Namun kecepatan *scanning* akan menjadi lambat untuk jumlah data yang besar.

Terdapat beberapa algoritma baru yang terbentuk untuk mengatasi kekurangan dari algoritma apriori. Salah satunya adalah algoritma *FP-Growth*. Algoritma *FP-Growth* tidak membangkitkan *candidate generation* dalam prosesnya dan hanya melakukan dua kali *scanning* data saja sehingga waktu yang dibutuhkan semakin singkat dan ruang penyimpanan yang dibutuhkan juga semakin sedikit. Di Jurusan Statistika ITS, pernah dilakukan penelitian mengenai *market basket analisis* dengan menggunakan algoritma *squeezer*. Berbeda dengan dua algoritma sebelumnya yang merupakan metode yang digunakan dalam *association rule*, *squeezer* adalah salah satu algoritma yang digunakan pada metode *clustering*. Algoritma *squeezer* memiliki kelebihan yaitu mampu menghasilkan pengelompokkan yang baik, memiliki stabilitas yang baik serta mampu menangani dimensi data besar secara efektif.

Terdapat beberapa penelitian sebelumnya yang telah menggunakan metode *market basket analysis* baik menggunakan algoritma *FP-Growth*, *apriori*, maupun *squeezer*. Failasufa (2014) pada penelitiannya menggunakan algoritma *apriori* memberikan hasil bahwa terdapat beberapa *rule* yang dihasilkan, pola pembelian dari ketiga cabang Supermarket Pamela hampir sama, dan nilai *confidence* tertinggi yang dihasilkan adalah sebesar 73,33%. Porkodi & P., (2015) dalam penelitiannya dengan menggunakan enam algoritma yaitu *apriori*, *aprioritid*, AIS, SETM, *Apriori hybrid* dan *FP-growth* memberikan hasil bahwa dari enam algoritma yang digunakan, berdasarkan *speed*, *apriori hybrid* sama bagusnya dengan *FP-Growth*, namun berdasarkan akurasi *FP-Growth* adalah yang terbaik. Hasil perbandingan dari enam metode memperlihatkan bahwa algoritma *FP-Growth* lebih cocok untuk memperoleh *association* yang signifikan dari *dataset* yang besar. Penelitian lain dilakukan oleh Fatihatul, Setiawan, & Rosadi (2011) memberikan hasil bahwa pada implementasi yang telah dilakukan menghasilkan nilai *strong confidence* paling tinggi 92%. Samuel (2007) pada penelitiannya menggunakan *FP-Growth* memberikan hasil

bahwa *FP-tree* yang terbentuk dapat menempatkan data transaksi yang memiliki item yang sama sehingga penggunaan memori komputer lebih sedikit dan proses pencarian *frequent itemset* lebih cepat. Yusuf & Kuncoro (2014) pada penelitiannya menerapkan algoritma *apriori* pada *alfamart* dan mendapatkan hasil suatu *layout* baru untuk penempatan *item* yang ada di *alfamart*. Susana, Limanto (2006) dalam penelitiannya menggunakan *Squeezer* memberikan hasil bahwa *clustering* data *market basket* membantu pengelompokkan produk-produk yang sering kali dibeli oleh konsumen secara bersama-sama dalam satu *cluster*. Berdasarkan beberapa penjelasan yang telah disebutkan, maka pada penelitian ini peneliti akan membandingkan algoritma *FP-growth*, *Apriori*, dan *Squeezer* pada analisis pola perilaku konsumen di minimarket *KI mart ITS*.

1.2 Rumusan Masalah

Di era modern seperti saat ini, terdapat banyak usaha ritel di Indonesia. Hal ini menyebabkan persaingan dalam merebut pangsa pasar menjadi semakin ketat. Konsumen menjadi semakin mudah untuk berpindah-pindah dari satu ritel ke ritel yang lain untuk menemukan ritel manakah yang lebih dirasa cocok. *KI mart ITS* sebagai minimarket yang baru terjun di dunia peritelan harus bisa membuat strategi pemasaran yang tepat agar mampu bersaing dalam merebut pangsa pasar sehingga tidak tereliminasi dan gulung tikar. Pada saat ini perkembangan teknologi yang semakin meningkat menyebabkan keberadaan data yang ada juga semakin banyak. Salah satu contoh adanya data yang menumpuk namun kurang begitu termanfaatkan dengan maksimal yaitu data transaksi *KI mart ITS*. Padahal dari data transaksi yang ini bisa digali untuk menghasilkan suatu informasi yang bermanfaat. Olehkarena itu permasalahan utama yang akan dibahas dalam penelitian ini adalah bagaimanakah perbandingan algoritma *FP-growth*, *Apriori*, dan *Squeezer* pada analisis pola perilaku konsumen di minimarket *KI mart ITS*.

1.3 Tujuan Penelitian

Berdasarkan perumusan masalah di atas, maka tujuan yang ingin dicapai dalam penelitian ini adalah mampu mendapatkan pola pembelian konsumen pada minimarket *KImart ITS* menggunakan algoritma *FP-growth*, *Apriori*, dan *Squeezer*.

1.4 Manfaat Penelitian

Manfaat yang didapatkan dari penelitian ini adalah dapat digunakan sebagai dasar dalam menyesuaikan promosi, pengaturan toko, membuat *layout* untuk situs *e-commerce*, mengontrol persediaan barang berdasarkan permintaan, mengurangi produk dengan *brand* tertentu yang dirasa tidak menguntungkan dan menghilangkan produk yang tidak sesuai dengan kebutuhan pelanggan minimarket *KImart ITS*. Manfaat lain yang dapat diperoleh adalah mampu mengetahui perbedaan algoritma *FP-growth*, *Apriori*, dan *Squeezer* saat diimplementasikan pada data transaksi.

1.5 Batasan Penelitian

Batasan yang digunakan penulis dalam penelitian ini adalah pada penelitian ini peneliti hanya melakukan analisis sampai ditemukan pola pembelian konsumen saja, tidak sampai pada tahap menentukan keputusan strategi pemasaran yang tepat.

(halaman ini sengaja dikosongkan)

BAB II TINJAUAN PUSTAKA

2.1 *Preprocessing*

Database yang ada sekarang ini kebanyakan merupakan data kotor, kebanyakan masih terdapat *missing* data, *noise*, dan data yang tidak konsisten didalamnya. Kualitas data yang rendah ini akan menghasilkan hasil yang berkualitas rendah juga. Olehkarena itu perlu adanya *preprocessing* untuk meningkatkan efisiensi pada proses yang dilakukan (Han & Kember, 2006). Saat menggunakan data *warehouse* untuk melakukan analisis, maka data yang digunakan akan sangat besar. Menganalisis data yang kompleks dengan jumlah yang besar akan membutuhkan waktu yang lama. Hal ini sangat tidak praktis dan tidak efisien. Teknik reduksi data dapat diaplikasikan untuk mengurangi volume data. Reduksi data bisa digunakan untuk mengurangi jumlah variabel ataupun jumlah objek penelitian. Pada kasus ini reduksi data yang digunakan bertujuan untuk mengurangi jumlah variabel penelitian. Proses reduksi dimensi yang dilakukan pada data transaksi dilakukan pada saat pemindaian pertama. Pada pemindaian pertama, akan dihitung nilai *support* untuk item dengan rumus sebagai berikut (Han & Kember, 2006).

$$\text{support}(X \Rightarrow Y) = P(XUY) \quad (2.1)$$

Semua item yang muncul pada transaksi yang lebih kecil daripada angka minimum *support* yang ditetapkan disebut sebagai *infrequent item*. Selanjutnya *infrequent item* akan dibuang dari transaksi sehingga item tersebut tidak akan menjadi bagian dari *itemset*. Item dalam setiap transaksi akan diurutkan secara menurun berdasarkan frekuensi dalam *database*. Meskipun algoritma tidak tergantung pada order yang spesifik, namun pada percobaan yang dilakukan dengan mengurutkan dengan urutan menurun memperlihatkan bahwa waktu pemrosesan menjadi lebih cepat daripada dengan membuat urutan random. Dalam percobaan yang telah dilakukan juga memperlihatkan bahwa

dengan urutan meningkat, proses yang dilakukan lebih lambat lagi dan ini lebih buruk daripada menyusun dengan urutan *random*. Pada hal ini kebiasaan yang dimiliki algoritma *FP-growth* sangat berkebalikan dengan algoritma Apriori, dimana pada pengaplikasian yang biasanya dilakukan akan menghasilkan waktu *running* yang lebih cepat saat disusun dengan urutan meningkat. Tetapi algoritma *FP-growth* ini sama dengan Eclat yang juga akan lebih menguntungkan jika menyusun urutan secara menurun (Borgelt).

2.2 *Assosiation Rule*

Assosiation rule merupakan salah satu metode yang digunakan dalam data mining. Tujuan dari *assosiation rule* adalah untuk menemukan *joint value* item-item yang sering muncul secara bersama-sama dalam suatu *database*. *Assosiation rule* biasanya diaplikasikan pada data biner $I_j \in \{0,1\}$, dimana biasanya digunakan dalam *market basket analysis*. Pada kasus ini observasi yang digunakan adalah data transaksi yang ada pada supermarket. Misalkan $I = \{I_1, I_2, \dots, I_m\}$ merupakan kumpulan item. Sekumpulan item disebut sebagai *itemset*. *Itemset* yang mengandung k -item disebut k -*itemset*. D merupakan *database* yang terdiri dari kumpulan transaksi, dimana $T \subseteq I$. Setiap transaksi memiliki ID transaksi yang disebut TID. Misalkan X berisi sekelompok item. Transaksi T dikatakan mengandung X jika dan hanya jika $X \subseteq T$. Sebuah *assosiation rule* dituliskan dalam bentuk $X \Rightarrow Y$, dimana $X \subset I, Y \subset I$, dan $X \cap Y = \emptyset$. *Association rule* berbentuk “*If antecedent then consequent*” (Larose & Larose, 2015). *Support rule* X dan Y disebut *antecedent* (*LHS: left hand side*) dan *consequent* (*RHS:right hand side*) dari *rule*. Terdapat dua proses yang dilakukan untuk mendapatkan *assosiation rule*.

1. Mendapatkan semua *frequent itemset*. Jika nilai *support* dari item I memenuhi nilai *minimum support* (*min_sup*) maka I adalah *frequent itemset*. Pada kasus ini algoritma yang digunakan untuk mendapatkan *frequent itemset* adalah *FP-Growth*.

2. Menghasilkan *association rule* yang kuat dari *frequent itemset*. *Assosiation rule* akan dikatakan kuat jika memenuhi *minimum support (min_sup)* dan *minimum confidence (min_conf)*. Nilai *confidence* dapat dihitung dengan menggunakan rumus sebagai berikut (Han & Kember, 2006).

$$confidence(X \Rightarrow Y) = P(Y|X) \quad (2.2)$$

Support merupakan persentase transaksi yang mengandung X dan Y. Sedangkan *confidence* merupakan persentase tingkat keyakinan pelanggan yang membeli X akan membeli Y. Misalkan *assosiation rule* yang didapatkan adalah $X \Rightarrow Y$ (*support*=20%, *confidence*=60%). Nilai *support* untuk rule $X \Rightarrow Y$ adalah 20% artinya 20% dari semua transaksi yang dianalisis, pelanggan akan membeli X dan Y secara bersamaan. Nilai *confidence* $X \Rightarrow Y$ adalah 60% , artinya 60% pelanggan yang membeli X juga akan membeli Y. *Assosiation rule* akan dikatakan menarik jika telah memenuhi nilai *minimum support* dan *minimum confidence*. Selanjutnya kriteria *interengeness* lain yang digunakan adalah nilai *lift*. *Lift* merupakan alat ukur yang baik untuk melihat seberapa baik *rule* yang dihasilkan. *Rule* yang dihasilkan dikatakan baik jika nilai *lift* lebih besar dari 1 (Berry & Linoff, 2004). Berikut ini adalah rumus untuk menghitung nilai *lift*.

$$lift(X \Rightarrow Y) = \frac{P(XUY)}{P(X)P(Y)} \quad (2.3)$$

2.3 Market Basket Analysis

Market basket analysis merupakan suatu proses untuk menganalisis kebiasaan pembelian pelanggan dengan menemukan hubungan antara item-item yang dibeli pelanggan dalam keranjang belanjanya. Informasi yang ditemukan nantinya dapat digunakan untuk mengembangkan strategi pemasaran dalam supermarket sehingga dapat meningkatkan keuntungan yang didapatkan. Sebagai contoh manager dari supermarket ingin mengetahui tentang kebiasaan pembelian pelanggannya. Sehingga

akan muncul pertanyaan barang apasajakah yang sering dibeli pelanggan secara bersama-sama. Untuk menjawab pertanyaan ini, maka kita dapat menggunakan *market basket analysis* (analisis keranjang belanjaan) pada data transaksinya. Selanjutnya kita bisa menggunakan hasilnya untuk merencanakan strategi pemasaran, strategi pengiklanan, atau membuat *design layout* toko yang baru. Salah satu strategi yang dapat diterapkan yaitu item-item yang sering dibeli secara bersamaan seperti sikat gigi dan pasta gigi dapat diletakkan berdekatan, agar jika pelanggan membeli sikat gigi maka pembeli juga akan teringat untuk membeli pasta gigi. Dengan meletakkan item-item yang sering dibeli secara bersamaan maka diharapkan akan meningkatkan jumlah penjualan item-item tersebut. *Market basket analysis* juga dapat membantu manager supermarket untuk menentukan barang yang manakah yang akan diberikan *discount*. Dengan memberikan *discount* pada suatu barang makan akan meningkatkan penjualan barang tersebut dan juga barang-barang lain yang berhubungan (Han & Kember, 2006).

Market basket data dapat direpresentasikan dengan format biner seperti pada contoh berikut. Dimana setiap baris merupakan data setiap transaksi dan kolom menggambarkan item-itemnya. Format ini biasanya digunakan saat *association rule* digunakan dalam mengidentifikasi *frequent itemset* (Venkatachari, 2016).

Tabel 2.1 *Binary Database*

TID	Bread	Butter	Jam	Juices	Paper goods	Dairy products	Canned goods
T1	1	1	0	1	0	1	1
T2	0	0	1	1	1	0	1
T3	1	0	1	0	1	0	1
T4	1	1	1	0	1	0	0
T5	0	1	0	1	1	1	0

Tabel 2.1 menunjukkan data transaksi. Seperti yang telah disebutkan bahwa baris menunjukkan setiap transaksi yang dilakukan dan kolom menunjukkan item-item yang ada dalam *database* transaksi. Apabila transaksi mengandung item dalam

kolom maka akan diberi nilai 1 namun jika tidak maka akan diberi nilai 1. Misalnya saja yaitu pada transaksi pertama pelanggan membeli *bread*, membeli *butter*, tidak membeli *jam*, membeli *juice*, tidak membeli *paper goods*, membeli *diary products*, membeli *canned goods*.

Tabel 2.2 Database Transaksi

TID	Item Set
T1	<i>Bread, butter, juice, dairy products, canned food</i>
T2	<i>Jam, juices, paper goods, canned goods</i>
T3	<i>Bread, jam, paper goods, canned goods</i>
T4	<i>Bread, butter, jam, paper goods</i>
T5	<i>Butter,juices, paper goods, dairy products</i>

Tabel 2.2 menunjukkan ID transaksi (TID) dan keterangan tentang item apasajakah yang dibeli pada transaksi ID tersebut. Misalnya yaitu transaksi dengan ID 1 membeli *bread, butter, juice, dairy products, canned food*. Dari *database* inilah patokan/acuan yang digunakan untuk membuat *binary database* pada Tabel 2.1.

2.4 *FP-Growth*

Frequent-pattern growth atau yang biasa disingkat *FP-growth* adalah suatu metode yang menghasilkan *frequent itemset* secara lengkap tanpa membangkitkan *candidate generation*. Langkah pertama yang dilakukan adalah membuat *database* menjadi suatu *frequent-pattern tree* atau *FP-tree*. Secara umum *FP-tree* adalah suatu pohon dengan ketentuan sebagai berikut.

1. Akan ada satu akar berlabel “null”, sekelompok item awal sebagai *subtrees* merupakan cabang dari akar dan merupakan *frequent item header* tabel.
2. Setiap titik pada item awal *subtree* terdiri dari tiga bagian: nama item, *count*, dan *node-link*, dimana nama item yang terdaftar ditampilkan pada titik item, *count* dari transaksi menampilkan jumlah tiap bagian yang mencapai titik ini, dan *node-link* menghubungkan titik selanjutnya pada *FP-tree* yang membawa nama item yang sama yang ditandai dengan garis panah putus-putus..

3. Setiap memasukkan pada *frequent-item-header* pada tabel berisi dua bagian yaitu nama item dan kepala dari *node-link* (sebuah poin pada titik pertama pada *FP-tree* membawa nama item).

Dalam membangun suatu struktur data yang padat dapat dibentuk dengan langkah-langkah sebagai berikut.

1. Menyeleksi data transaksi untuk menemukan *frequent itemset*.
2. Jika beberapa transaksi berbagi *frequent itemset*, maka hal ini memungkinkan menggabungkan kelompok menjadi jumlah kejadian sebagai *count*. Akan mudah untuk mengecek apakah dua kelompok identik yaitu jika *frequent item* pada semua transaksi tercatat dengan order tetap.
3. Jika dua transaksi berbagi awalan umum, berdasarkan beberapa urutan *frequent item*, bagian yang dibagi dapat disatukan menggunakan satu struktur awalan sepanjang *count* yang terdaftar. Jika *frequent item* diurutkan menurun berdasarkan jumlah frekuensi, maka akan ada kemungkinan membagi struktur awal lebih banyak.

Misalkan tabel berikut ini merupakan *database* transaksi suatu supermarket (data asli, belum dilakukan reduksi dimensi) ditampilkan dalam Tabel di bawah dengan nilai minimum support yang digunakan adalah 3.

Tabel 2.3 *Database* Transaksi Sebelum Reduksi Dimensi

TID	Item yang dibeli
1	f,a,c,d,g,i,m,p
2	a,b,c,f,l,m,o
3	b,f,h,j,o
4	b,c,k,s,p
5	a,f,c,e,l,p,m,n

Pertama-tama akan dilakukan *scanning database* untuk melihat nilai frekuensi setiap *item*. Frekuensi dari item-item yang ada pada database yaitu $\langle (f:4), (c:4), (a:3), (b:3), (m:3), (p:3), (o:2), (l:2), (d:1), (e:1), (g:1), (h:1), (i:1), (j:1), (k:1), (n:1), (s:1) \rangle$, dimana angka setelah tanda “:” mengindikasikan nilai *support count*. Setiap item pada tabel diurutkan dengan urutan menurun.

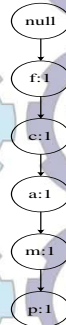
Pola urutan ini penting sebab setiap bagian dari pohon akan mengikuti pola ini. Item dengan nilai *support* kurang dari *min_support* yang ditentukan akan dikeluarkan dari pengamatan. Item-item yang memenuhi *min_support* yaitu $\langle (f:4), (c:4), (a:3), (b:3), (m:3), (p:3) \rangle$. Berikut ini adalah kemunculan item yang *frequent* dalam setiap transaksi yang terbentuk setelah mengeluarkan item yang tidak memenuhi *min_support* dan diurutkan sesuai frekuensi.

Tabel 2.4 Database Transaksi Setelah Reduksi Dimensi

TID	Frequent itemset
1	f,c,a,m,p
2	f,c,a,b,m
3	f,b,
4	c,b,p
5	f,c,a,m,p

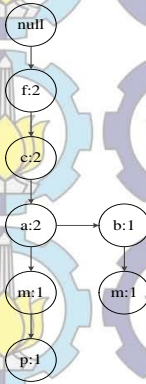
Setelah didapatkan daftar transaksi yang hanya berisi item yang *frequent*, langkah selanjutnya yaitu melakukan *scanning* yang kedua pada transaksi untuk pembentukan *FP-tree*. Akar dari pohon ini akan diberikan label “null”.

1. Hasil pemindaian transaksi yang pertama digunakan untuk membangun cabang yang pertama pada pohon: $\{(f:1), (c:1), (a:1), (m:1), (p:1)\}$. *Frequent item* pada transaksi yang ditulis berdasarkan urutan pada list *frequent item*. Berikut ini adalah gambar yang mengilustrasikan pembentukan *FP-tree* setelah pemindaian pada TID 1.



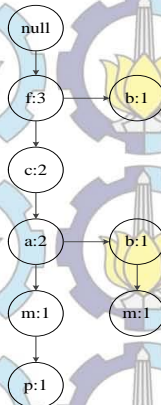
Gambar 2.1 *FP-tree* Setelah Pemindaian TID 1

2. Untuk transaksi kedua, karena urutan *frequent item* $\langle f, c, a, b, m \rangle$ berbagi awalan $\langle f, c, a \rangle$ dengan bagian yang sudah ada $\langle f, c, a, m, p \rangle$ maka *count* untuk setiap titik sepanjang awalan akan meningkat 1, dan titik baru (b:1) dibentuk dan disambungkan sebagai anak dari (a:2) dan titik baru yang lain (m:1) dibentuk dan disambungkan dengan anak dari (b:1). Gambar berikut mengilustrasikan *FP-tree* setelah pemindaian TID 2.



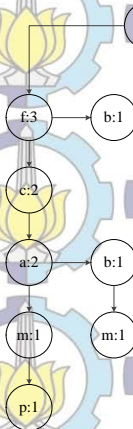
Gambar 2.2 *FP-tree* Setelah Pemindaian TID 2

3. Untuk transaksi ketiga, karena *frequent item* yang tercatat $\langle f, b \rangle$ hanya berbagi $\langle f \rangle$ dengan *f* awalan *subtree*, maka *count* *f* naik 1 dan titik baru (b:1) dibentuk dan disambungkan sebagai anak cabang dari (f:3). Sehingga *FP-tree* setelah pemindaian TID 3 dapat diilustrasikan dengan gambar sebagai berikut



Gambar 2.3 *FP-tree* Setelah Pemindaian TID 3

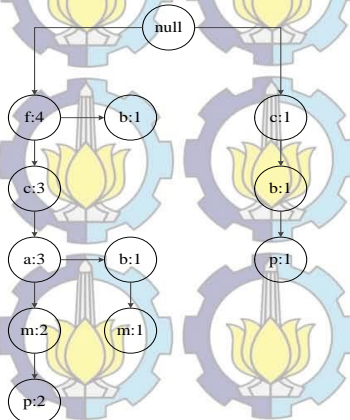
4. Pemindaian pada transaksi keempat membentuk cabang kedua pada pohon $\langle (c:1), (b:1), (p:1) \rangle$. Setelah pemindaian TID 4, ilustrasi *FP-tree* dapat dilihat pada gambar berikut.



Gambar 2.4 *FP-tree* Setelah Pemindaian TID 4

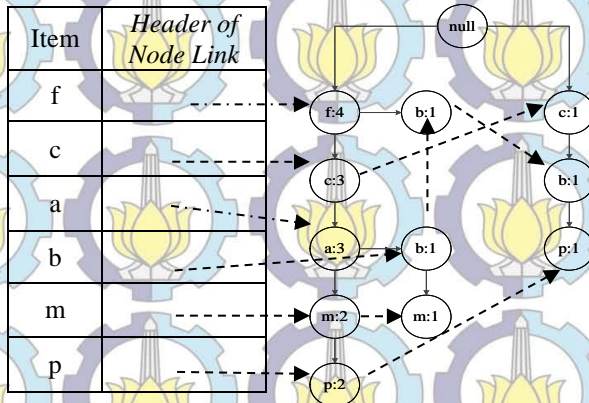
5. Pada transaksi terakhir, karena *frequent item* $\langle f, c, a, m, p \rangle$ identik dengan yang pertama maka bagian ini berbagi *count* pada setiap titik sehingga nilainya bertambah 1. Sehingga

FP-tree yang dapat terbentuk setelah pemindaian TID 5 adalah sebagai berikut.



Gambar 2.5 *FP-tree* Setelah Pemindaian TID 5

Untuk memfasilitasi pohon transversal, suatu kepala item pada tabel dibangun dengan setiap poin pada item dihubungkan dengan kejadian pertama yang muncul pada pohon. Titik dengan nama item yang sama akan dihubungkan dengan urutan titik yang tersambung. Setelah melakukan pemindaian pada semua transaksi, maka pohon yang terbentuk dari semua titik-titik yang terhubung tersebut akan terbentuk seperti pada gambar berikut.



Gambar 2.6 Pembentukan *FP-tree*

Berikut ini adalah algoritma untuk pembuatan *FP-tree*.

Input : *database* transaksi dengan nilai *minimum support* tertentu.

Output : *FP-tree*

Metode : *FP-tree* dibentuk dengan cara sebagai berikut.

1. Melakukan pemindaian pada *database* transaksi sekali. Mengumpulkan F, sekelompok *frequent item* dan *support* untuk setiap item. Mengurutkan F dengan urutan *support* menurun sebagai *Flist*.
2. Membuat akar dari *FP-tree*, T, dan memberi label “null”. Untuk setiap transaksi pada *database* akan melakukan tahapan seperti berikut.

Memilih *frequent item* di Trans dan mengurutkan berdasarkan *Flist*. *Frequent item* pada daftar yang telah diurutkan di Trans menjadi [p|P], dimana p merupakan elemen pertama dan P adalah sisanya yaitu *insert_tree*([p|P], T). Fungsi *insert_tree* ([p|P], T) dilakukan jika T memiliki N sehingga $N.item-name = p.item-name$, maka tambahkan 1 pada *count* N; jika tidak maka membuat *node* N baru, dengan *count* inisial yaitu 1, link induknya dihubungkan dengan T, dan *node-link* dihubungkan dengan

node pada *item-name* yang sama dengan struktur *node-link*. Jika P tidak kosong, maka panggil *insert_tree*(P, N) (Han, Pei, Yin, & Mau, 2004).

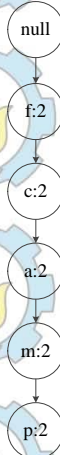
Untuk *frequent item* a_i , semua pola mungkin berisi hanya satu *frequent item* dan a_i dapat diperoleh dengan mengikuti *node link* a_i , mulai dari kepala a_i pada *FP-tree header*. Sifat ini secara langsung berasal dari proses pembentukan *FP-tree*, dan hal ini memfasilitasi semua informasi *frequent-pattern* yang berhubungan dengan a_i dengan memotong *FP-tree* yang mengikuti *node-links* a_i . Semua pola yang berisi *frequent item* yang berisi a_i dapat dikumpulkan dengan memulai *node-link head* a_i dan mengikuti *node-link* nya. Proses penggalian informasi dimulai dari *node-link header table* yang paling bawah. Berikut ini adalah langkah pembentukan *FP-Growth* (Samuel, 2007).

1. Membangun *conditional pattern base*. *Conditional pattern base* bisa didapatkan melalui *FP-tree* yang terbentuk. Caranya yaitu dengan melihat *FP-tree* yang berisi akhiran a_i . Setiap lintasan yang tidak mengandung a_i dibuang. Jadi *support count* untuk setiap titik adalah nilai *support count* untuk titik tersebut yang muncul bersama a_i . Dalam menyebutkan *conditional pattern base*, hanya *prefix* nya saja yang disebutkan.
2. Membangun *conditional FP-tree*. Pada tahap ini *support count* dari setiap *item* pada *conditional pattern base* dijumlahkan. Setiap *item* yang memiliki *support count* lebih dari minimal *support* maka akan dibangun *conditional FP-tree* nya.
3. Pencarian *frequent itemset*. Jika *conditional FP-tree* merupakan *single path*, maka untuk mendapatkan *frequent itemset* dilakukan kombinasi untuk setiap *item* yang berada pada *conditional FP-tree*. Jika *conditional FP-tree* bukan merupakan *single path* maka *frequent itemset* dibangun secara rekursif.

Berikut ini adalah pembentukan *FP-Growth* untuk contoh sebelumnya.

1. Pembentukan *FP-Growth* untuk *suffix p*

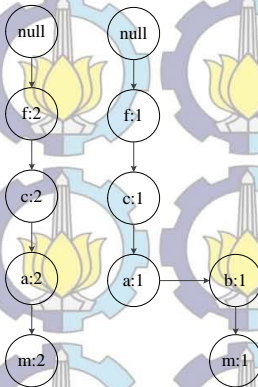
Berikut ini adalah *conditional pattern base* untuk *suffix p*. Dimana untuk semua lintasan yang tidak mengandung *p* nilai *support count* sudah dikurangi. Terdapat dua *conditional pattern base* dengan *suffix p* yaitu (fca:2) dan (cb:1). Agar lebih mudah dalam mendapatkan *conditional FP-tree* untuk *suffix p*, perhatikan gambar berikut ini.



Gambar 2.7 *Conditional Pattern Base* untuk *Suffix p*

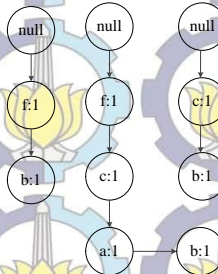
Berdasarkan Gambar 2.7 *conditional pattern base* yang memenuhi *minimal support* adalah item *c* yaitu dengan *support count* 3. Sehingga didapatkan *conditional FP-tree* adalah $\{(c:3)\}p$. Pada *suffix p*, *conditional FP-tree* yang terbentuk merupakan *single path* sehingga untuk mendapatkan *frequent itemset* dilakukan dengan mengkombinasikan saja *conditional FP-tree*. Jadi *frequent itemset* yang terbentuk adalah *p, cp*.

2. Setelah menghilangkan *item* yang tidak mengandung *m* dalam lintasan, maka *conditional pattern base* yang terbentuk adalah (fca:2) dan (fcab:1). Berikut ini adalah gambar untuk *conditional pattern* yang terbentuk.



Gambar 2.8 *Conditional Pattern Base* untuk Suffix m

Berdasarkan gambar 2.8, item yang memiliki *support count* lebih dari atau sama dengan 3 adalah a, c, f. Sehingga *conditional FP-tree* yang terbentuk adalah $\{(f:3, c:3, a:3)\}m$. *Conditional FP-tree* yang terbentuk merupakan *single path*, sehingga *frequent itemset* didapatkan dengan mencari kombinasinya. *Frequent itemset* yang terbentuk adalah m, fm, cm, am, fcm, fam, cam, fcam.



Gambar 2.9 *Conditional Pattern Base* untuk Suffix b

Pada gambar 2.9 dapat dilihat bahwa *support count* untuk item f adalah 2 dan untuk c adalah 1. Karena *support count* dari f dan c tidak lebih dari 3 maka untuk suffix b tidak ada *conditional FP-tree* yang dibangun. Tidak ada *conditional FP-tree* yang dibangun, sehingga *frequent itemset* yang terbentuk hanya b saja.

3. Pembentukan *FP-Growth* untuk *suffix a*
Conditional pattern base yang terbentuk untuk *suffix a* hanya ada satu yaitu (fc:3). Berikut ini adalah gambar *conditional pattern base* yang terbentuk untuk *suffix a*.



Gambar 2.10 *Conditional Pattern Base* untuk *Suffix a*

Conditional pattern base untuk *suffix a* berbentuk *single path* maka semua item diatas pasti memenuhi *minimal support count* karena *a* memenuhi *minimal support count*. Jadi *conditional FP-tree* yang terbentuk adalah {(f:3, c:3)}|a. Pada *suffix b*, *conditional FP-tree* yang terbentuk juga merupakan *single path*. Jadi *frequent itemset* yang terbentuk adalah a, fa, ca, dan fca.

4. Pembentukan *FP-Growth* untuk *suffix c*
 Sama halnya dengan *suffix a*, pada *suffix c* *conditional pattern base* yang terbentuk hanya ada satu yaitu (f:3). Untuk *suffix c*, *conditional pattern base* yang terbentuk adalah seperti pada gambar berikut.



Gambar 2.11 *Conditional Pattern Base* untuk *Suffix c*

Sama halnya dengan *suffix a*, pada *suffix b* conditional pattern base juga merupakan single path. Jadi *conditional FP-tree* yang terbentuk adalah $\{(f:3)\}c$. *Frequent itemset* yang terbentuk untuk *suffix c* adalah *c* dan *fc*.

5. Pembentukan *FP-Growth* untuk *suffix f* conditional pattern base

Berbeda dengan *suffix* yang lainnya, pada *suffix f* tidak ada *conditional pattern base* yang terbentuk. Karena tidak ada *conditional pattern base* yang terbentuk maka tidak ada *conditional FP-tree* yang terbentuk dan *frequent itemset* hanya berisi dirinya sendiri yaitu *item f*.

Hasil dari *conditional pattern-bases*, *conditional FP-tree* dan *frequent itemset* yang didapatkan dirangkum dalam Tabel berikut.

Tabel 2.5 Conditional Pattern-bases, Conditional FP-tree dan Frequent Itemset yang Terbentuk

Suffix	Conditional Pattern-base	Conditional FP-tree	Frequent Itemset
P	$\{(f\ cam:2), (cb:1)\}$	$\{(c:3)\}p$	p, cp
m	$\{(f\ ca:2), (fcab:1)\}$	$\{(f:3, c:3, a:3)\}m$	m, fm, cm, am, fcm, cam, fcam
b	$\{(f\ ca:1), (f:1), (c:1)\}$	\emptyset	b
a	$\{(fcc:3)\}$	$\{(f:3, c:3)\}a$	a, fa, ca, fca
c	$\{(f:3)\}$	$\{(f:3)\}c$	c, fc
f	\emptyset	\emptyset	f

2.5 Apriori

Apriori adalah algoritma yang diperkenalkan oleh R. Agrawal dan R. Srikant pada tahun 1994 untuk mencari *frequent itemset* untuk data Boolean pada *association rule* (Han & Kember, 2006). *Apriori* menggunakan metode iteratif dimana *k-itemset* digunakan untuk menentukan $(k+1)$ *itemset*. Jika *itemset infrequent* maka supersethnyapun pasti *infrequent*. Langkah awal dalam algoritma apriori yang harus dilakukan adalah melakukan *scanning* data untuk menemukan *support* untuk setiap *1-itemset*.

Hanya item yang memenuhi minimum support saja yang akan dilakukan analisis lebih lanjut. Item yang memenuhi minimum support ini dinotasikan dengan L_1 . L_1 akan digunakan untuk menemukan L_2 . L_2 akan digunakan untuk menemukan L_3 , dan seterusnya sampai tidak ada k -itemset yang dapat dibentuk. Untuk menemukan L_k dibutuhkan satu kali *full scanning database*. Berikut ini adalah *step-step* dalam algoritma *apriori*.

1. *Join step*. Tahap ini dilakukan untuk membangkitkan *candidate generation* yang dinotasikan dengan C_k . Satu set kandidat k -itemset dibangkitkan dengan mengkombinasikan L_{k-1} dengan dirinya sendiri.
2. *Prune step*. C_k merupakan superset dari L_k dimana tidak semua anggotanya *frequent*, tetapi semua k -itemset yang *frequent* masuk ke dalam C_k . Pada tahap ini dilakukan *scanning database* lagi untuk menemukan *support count* dari semua kandidat C_k . Kemudian kandidat yang memenuhi minimum *support* akan disebut sebagai L_k . Untuk mengurangi ukuran C_k , jika $(k-1)$ subset dari k -itemset bukan bagian dari L_{k-1} maka kandidatnya pasti *infrequent* sehingga dapat dikeluarkan dari C_k .

Untuk lebih jelasnya, berikut ini adalah contoh penerapan algoritma *apriori*. Misalkan *support count* yang digunakan adalah 2. Berikut ini adalah tabel *database* transaksi.

Tabel 2.6 Database Transaksi

TID	Item yang dibeli
T1	A, B, E
T2	B, D
T3	B, C
T4	A, B, D
T5	A, C
T6	B, C
T7	A, C
T8	A, B, C, E
T9	A, B, C

Langkah pertama yaitu *scanning database* dan menghitung *support count* untuk *1-itemset*. $C_k = \{A, B, C, D, E\}$ Berikut ini adalah hasilnya.

Tabel 2.7 Support Count 1-Itemset

<i>1-Itemset</i>	<i>Support Count</i>
A	6
B	7
C	6
D	2
E	2

Karena semua item memenuhi minimum *support* maka $L_1 = \{A, B, C, D, E\}$. Selanjutnya membangkitkan *candidate generation*. $C_2 = \{AB, AC, AD, AE, BC, BD, BE, CD, CE, DE\}$. Setelah itu masuk ke tahap *prune* yaitu melakukan *scanning* kembali apakah subset untuk setiap itemset yang ada pada C_2 merupakan bagian dari L_1 . Jika tidak maka *itemset* ini akan dikeluarkan dari C_2 . Selanjutnya menghitung *support count* untuk C_2 . Berikut adalah hasilnya.

Tabel 2.8 Support Count 2-Itemset

<i>2-Itemset</i>	<i>Support Count</i>
AB	4
AC	4
AD	1
AE	2
BC	4
BD	2
BE	2
CD	0
CE	1
DE	0

Dari tabel di atas dapat dilihat bahwa terdapat tiga itemset yang tidak memenuhi minimum *support* yaitu CD, CE, dan DE sehingga item ini tidak termasuk ke dalam L_2 . Jadi $L_2 = \{AB, AC, AD, AE, BC, BD, BE, CD, CE, DE\}$. Kembali ke tahap *join step* yaitu membangkitkan *candidate generation* untuk *3-itemset*. Maka didapatkan $C_3 = \{ABC, ABE, ACE, BCD, BCE, BDE\}$.

Selanjutnya dilakukan *scanning* kembali pada *database*, ternyata ada subset dari C_3 yang bukan merupakan bagian dari L_2 , sehingga itemset tersebut dikeluarkan dari C_3 . Sehingga didapatkan $C_3 = \{ABC, ABE\}$. Berikut adalah tabel nilai *support count* untuk C_3 .

Tabel 2.9 *Support Count 3-Itemset*

3-Itemset	Support Count
ABC	2
ABE	2

Ternyata semua data untuk 3-itemset ini memenuhi *minimum support*, maka didapatkan $L_3 = \{ABC, ABE\}$ dan dapat dilanjutkan ke tahap selanjutnya yaitu membangkitkan *candidate generation* untuk 4-itemset. Maka $C_4 = \{ABCE\}$. Namun setelah dilakukan *scanning database* ternyata terdapat subset dari itemset ini yang tidak frequent maka $C_4 = \emptyset$ sehingga $L_4 = \emptyset$. Sehingga proses dihentikan.

2.6 Squeezer

Squeezer adalah salah satu algoritma yang digunakan dalam *clustering* data. Konsep algoritma *squeezer* adalah dengan minimum *similarity* dari satu barang dengan *cluster* yang ada, apabila nilai minimum *similarity* produk tersebut tidak memenuhi minimum *similarity* maka produk tersebut akan dibuatkan *cluster* baru. Berikut ini adalah langkah-langkah yang digunakan pada *squeezer* (Limanto, 2005).

1. Menentukan α, β dan nilai minimum *similarity*. Untuk mendapatkan minimum *similarity* bisa menggunakan rumus sebagai berikut.

$$S(A, B) = X^\alpha Y^\beta \quad (2.4)$$

$$X = \frac{N(A, B)^2}{N(A) \times N(B)} \quad (2.5)$$

$$\gamma = \frac{N(A,B)}{N_{Total}} \quad (2.6)$$

2. Membaca *record* pertama dari data yang kemudian dibentuk menjadi sebuah cluster.
3. Membaca *record* berikutnya dan menghitung *similarity* antara *record* yang dibaca dengan *cluster* yang terbentuk sebelumnya.
4. Mencari nilai *similarity* terbesar untuk *record* tersebut. Apabila nilai *similarity* terbesar memenuhi kriteria minimum *similarity* maka *record* ini akan masuk ke dalam *cluster* tersebut, namun jika tidak maka akan membentuk *cluster* baru.
5. Mengulangi langkah ke 3 dan 4 sampai semua *record* terbaca.

Berikut ini adalah contoh penggunaan metode *squeezer*. Digunakan nilai $\alpha = 1$, $\beta = 1$, dan nilai minimum *similarity* yang digunakan adalah 0,05. Berikut ini adalah tabel data transaksi penjualan.

Tabel 2.10 Transaksi Penjualan

No. Transaksi	Kode Barang yang terjual
100	1,3,4
200	2,3,5
300	1,2,3,5
400	2,5

Berdasarkan tabel transaksi di atas didapatkan tabel barang yang selanjutnya dibaca sebagai *record* adalah seperti pada tabel berikut ini.

Tabel 2.11 Kode Barang

Kode Barang
1
2
3
4
5

Langkah pertama yaitu membaca *record* pertama dan menjadikannya sebagai *cluster 1*. Jadi *cluster 1* berisi kode barang 1. Selanjutnya membaca *record* kedua yaitu kode barang 2. Lalu menghitung nilai *similarity* kode barang 2 dengan *cluster 1*.

$$S(\text{Cluster1, kode barang} = 2) = \left(\frac{1^2}{2 \times 3} \right)^1 \times \left(\frac{1}{4} \right)^1 = 0,042$$

Nilai *similarity* yang didapatkan adalah 0,042. Nilai ini kurang dari minimum *similarity* yang ditetapkan maka barang 2 dibuatkan *cluster* baru yaitu *cluster 2*. Selanjutnya membaca *record 3* dan menghitung *similarity* kode barang 3 untuk *cluster 1* dan *cluster 2* seperti berikut.

$$S(\text{Cluster1, kode barang} = 3) = \left(\frac{2^2}{2 \times 3} \right)^1 \times \left(\frac{2}{4} \right)^1 = 0,333$$

$$S(\text{Cluster2, kode barang} = 3) = \left(\frac{2^2}{3 \times 3} \right)^1 \times \left(\frac{2}{4} \right)^1 = 0,222$$

Nilai *similarity* yang digunakan adalah yang tertinggi yaitu *similarity* antara kode barang=3 dengan *cluster 1* yaitu sebesar 0,333. Karena nilai *similarity* ini lebih dari minimum *similarity* yang ditetapkan maka kode barang 3 masuk ke dalam *cluster 1*. Lalu membaca *record 4* dan menghitung *similarity* untuk *cluster 1* dan 2 seperti berikut.

$$S(\text{Cluster1, kode barang} = 4) = \left(\frac{1^2}{2 \times 1} \right)^1 \times \left(\frac{1}{4} \right)^1 = 0,125$$

$$S(\text{Cluster2, kode barang} = 4) = \left(\frac{0^2}{3 \times 1} \right)^1 \times \left(\frac{0}{4} \right)^1 = 0$$

Nilai *similarity* tertinggi adalah 0,125 yaitu *similarity* untuk kode barang 4 dan *cluster 1*. Nilai *similarity* yang dihasilkan lebih dari minimum *similarity* yang ditetapkan. Jadi kode barang 4 masuk ke dalam *cluster 1*. Selanjutnya yaitu membaca *record* terakhir. Nilai *similarity* kode barang 5 untuk kedua *cluster* yang ada adalah sebagai berikut.

$$S(\text{Cluster1, kode barang} = 5) = \left(\frac{0^2}{1 \times 3} \right)^1 \times \left(\frac{0}{4} \right)^1 = 0$$

$$S(Cluster2, kode\ barang = 5) = \left(\frac{3^2}{3 \times 3}\right)^1 \times \left(\frac{3}{4}\right)^1 = 0,75$$

Nilai *similarity* tertinggi adalah 0,75 dimana nilai ini adalah nilai *similarity* untuk kode barang 5 dengan *cluster* 2. Nilai 0,75 yang dihasilkan lebih dari minimum *similarity* yang ditetapkan sehingga kode barang 5 masuk ke dalam *cluster* 2.

Berdasarkan algoritma *squeezer* maka terbentuklah dua *cluster*. *Cluster* 1 beranggotakan kode barang 1,3, dan 4. Sedangkan *cluster* 2 beranggotakan sisanya yaitu 2 dan 5.

2.7 Interpretasi Hasil Association Rule

FP-Growth dan *apriori* adalah algoritma yang digunakan dalam *association rule*. Sehingga hasil yang didapatkan adalah suatu aturan yang biasanya disebut sebagai *rule*. Berikut ini adalah contoh *rule* yang terbentuk.

Tabel 2.12 Contoh *Rule*

No	LHS	RHS	Support	Confidence	Lift
1	1299 Teh Pucuk	256 Air Mineral	0,0035	0,1598	1,0524
2	1057 Roti	256 Air Mineral	0,0031	0,1751	1,1533

LHS (*left hand side*)/*antecedent/condition* adalah sebutan untuk item yang dibeli terlebih dahulu oleh konsumen. RHS (*right hand side*)/*consequent/result* adalah sebutan untuk barang yang dibeli selanjutnya oleh konsumen. LHS ataupun RHS bisa berisi lebih dari satu barang. Dalam melakukan analisis, barang yang dibeli konsumen biasanya disebut sebagai item.

Berdasarkan tabel 2.12 didapatkan dua *rule* yang terbentuk. Arti dari *rule* pertama yang terbentuk adalah terdapat 0,35% konsumen yang membeli item dengan kode 1299 (Teh Pucuk) dan item dengan kode 256 (Air Mineral) secara bersamaan. 15,98% konsumen yang membeli item 1299 (Teh Pucuk) juga akan membeli item 256 (Air Mineral). Nilai *lift* yang dihasilkan lebih dari satu artinya *rule* yang dihasilkan signifikan.

BAB III METODOLOGI PENELITIAN

3.1 Sumber Data

Data yang digunakan dalam penelitian ini adalah data sekunder yang didapatkan dari minimarket *Klmart*. Dimana data yang digunakan adalah data transaksi pada bulan 1 Maret 2016 sampai dengan 28 Maret 2016 yaitu sejumlah 9871 transaksi.

3.2 Variabel Penelitian

Variabel yang digunakan dalam penelitian ini adalah data jenis barang (item) yang dibeli oleh konsumen dan jumlah barang yang dibeli. Sedangkan untuk variabel harga tidak digunakan untuk melakukan analisis data. Perangkat keras yang digunakan dalam tugas akhir ini adalah laptop HP dengan *processor* Intel(R) Core(TM) i3-4030U CPU @ 1.90GHz 1.90 GHz, RAM 2 GB, dan *system type* 32-bit.

Pada analisis yang dilakukan akan digunakan dua jenis data yaitu data barang detail dan data barang umum. Data barang detail adalah data barang dengan keterangan nama barang yang masih lengkap yaitu terdapat *merk* barang, jenis makanan, jenis rasa, dan ukuran atau *netto* dari barang. Sedangkan untuk data barang umum adalah data barang detail yang telah diubah ke dalam bentuk yang lebih umum. Pada data barang umum keterangan ukuran atau *netto* dari barang dihilangkan.

3.3 Langkah Analisis

Beberapa tahapan yang dilakukan dalam analisis ini adalah sebagai berikut.

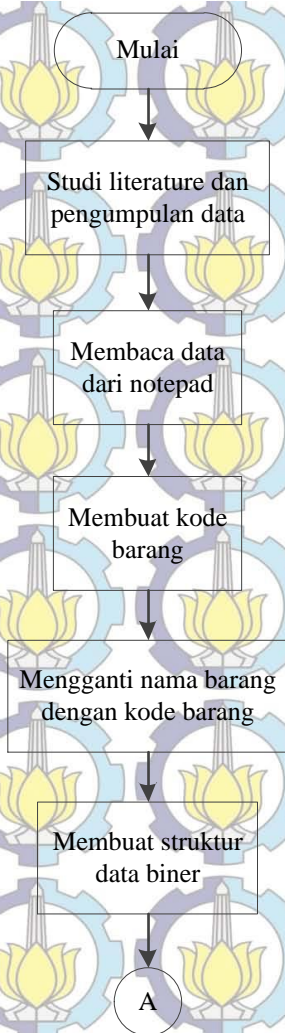
1. Studi *literature* dan mengumpulkan data transaksi yang ada di *Klmart*.
2. Melakukan *preprocessing data*. Beberapa tahapan *preprocessing data* yang dilakukan adalah.
 - a. Mentransformasi data mentah yang berbentuk *notepad* ke dalam bentuk data biner yang siap diolah dengan menggunakan software R. Pada saat membaca data struk

informasi yang diambil hanya informasi yang dibutuhkan saja seperti nama barang yang dibeli konsumen. Variabel harga dalam data transaksi tidak diambil sebab informasi mengenai harga disini tidak digunakan.

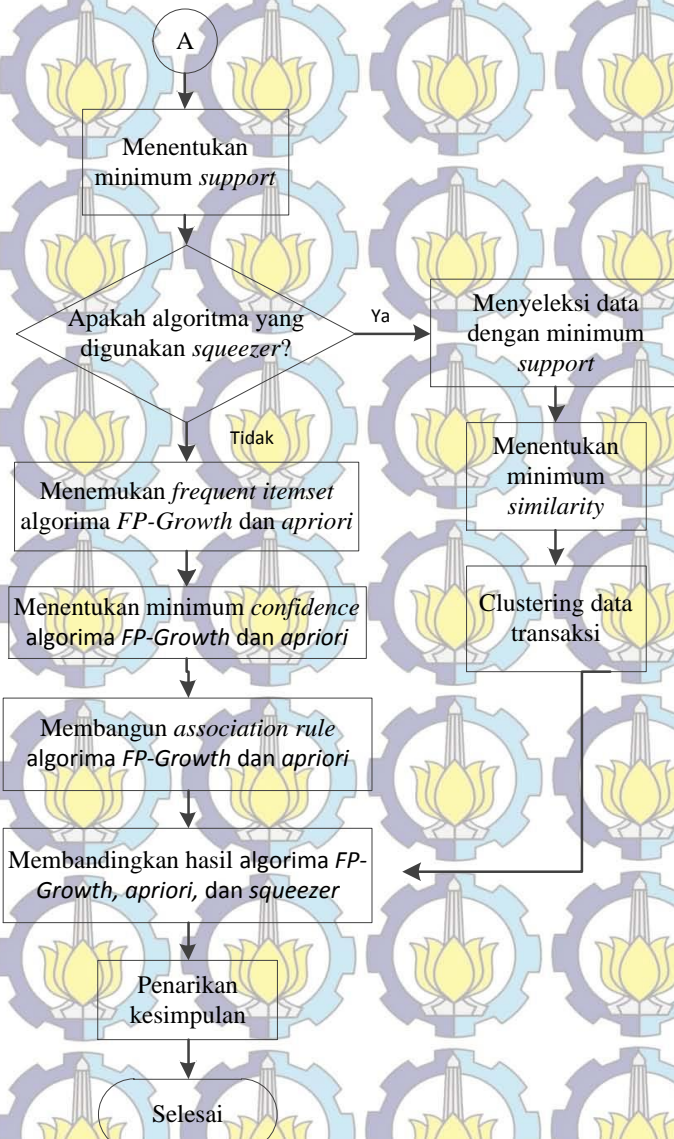
- b. Membuat kode barang untuk data dengan nama barang detail dan membuat kode barang untuk data dengan nama barang yang lebih umum.
- c. Mengganti nama barang dalam data transaksi dengan kode barang
- d. Mentransformasi data kedalam bentuk *binary database*.
3. Menyeleksi data dengan nilai minimum *support* untuk algoritma *squeezer*.
4. Melakukan pengelompokan barang yang sering dibeli secara bersama-sama dengan metode *clustering* dengan algoritma *squeezer*.
5. Melakukan pencarian *frequent itemset* berdasarkan algoritma *FP-Growth*, *Apriori*.
6. Menentukan nilai minimum *confidence* untuk algoritma *FP-Growth* dan *apriori*.
7. Dari *itemset* yang sering muncul, dicari *association rule* pada *itemset* yang memenuhi *minimum confidence* menggunakan algoritma *FP-Growth* dan *apriori*.
8. Membandingkan hasil dari algoritma *FP-Growth*, *apriori*, dan *squeezer*.
9. Membuat kesimpulan hasil analisis pada penelitian yang telah dilakukan.

3.4 Diagram Alir

Berdasarkan langkah analisis yang telah dilakukan, maka didapatkan diagram alir seperti pada gambar berikut.



Gambar 3.1 Diagram Alir



Gambar 3.1 Diagram Alir (Lanjutan)

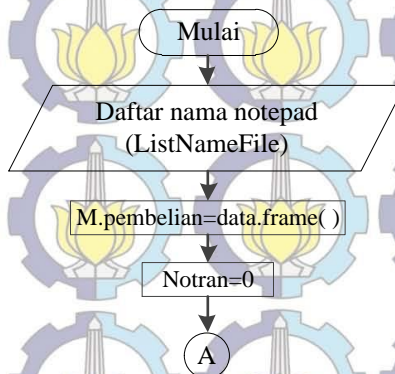
BAB IV ANALISIS DAN PEMBAHASAN

4.1 *Preprocessing*

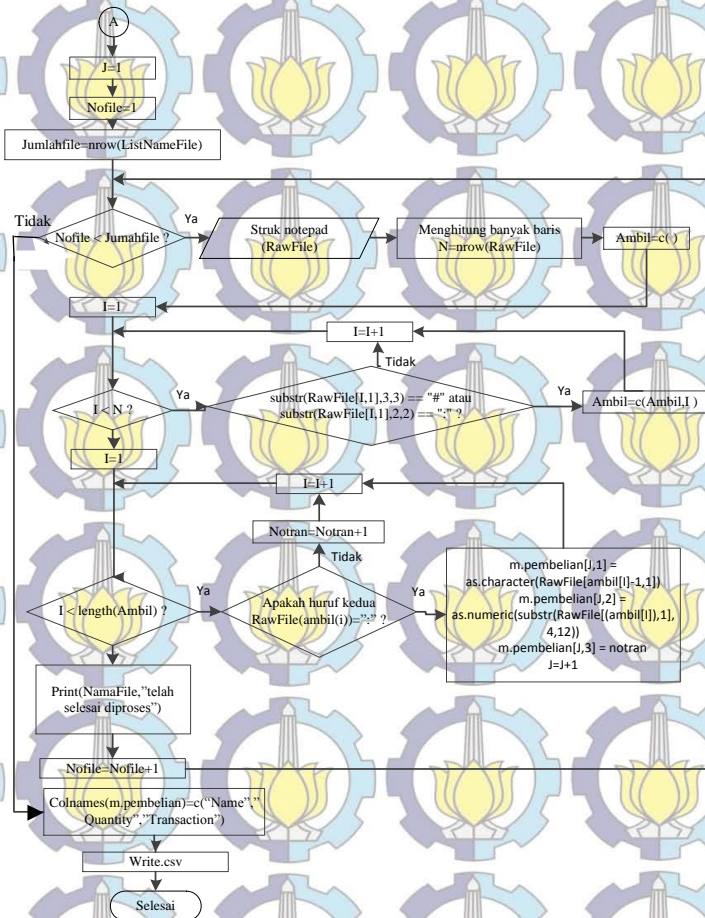
Data yang digunakan dalam penelitian ini adalah data sekunder yang didapatkan dari minimarket *Klmart*. Data yang digunakan adalah 9871 data transaksi yang ada di minimarket *Klmart*. Data yang telah tersedia belum bisa diolah. Olehkarena itu perlu dilakukan *preprocessing* terlebih dahulu.

4.1.1 Membaca Data dari Notepad

Data transaksi tersebut berupa struk notepad seperti yang terlihat pada lampiran 1. Olehkarena itu, data yang ada ini tidak bisa langsung diolah dan dianalisis. Tahap pertama dalam *preprocessing* yang dilakukan adalah dengan membuat *syntax* untuk membaca data dari notepad. Informasi yang akan diambil dari struk tersebut antara lain nama barang yang dibeli, jumlah barang yang dibeli, dan transaksi keberapakah barang tersebut dibeli. Kemudian informasi yang diambil ini akan disimpan dalam bentuk *csv*. Berikut ini adalah algoritma untuk membaca file struk dalam notepad.



Gambar 4.1 Flowchart Membaca File Notepad



Gambar 4.1 Flowchart Membaca File Notepad (Lanjutan)

Data struk yang tersedia adalah data harian. Jadi dalam satu notepad terdapat semua transaksi yang terjadi dalam satu hari. Olehkarena itu dibuat sebuah notepad yang berisi daftar nama notepad transaksi harian yang akan dibaca. Pada tahap pertama ini *output* yang dihasilkan adalah daftar pembelian konsumen yang berisi 3 kolom dan 20.262 baris. Kolom pertama adalah kolom *name* yang menunjukkan daftar nama barang yang dibeli konsumen. Kolom kedua menunjukkan jumlah barang yang dibeli dan kolom yang ketiga berisi pada transaksi keberapakah barang tersebut dibeli. Satu baris hanya berisi satu barang yang dibeli konsumen. Jadi dalam satu transaksi bisa terdiri dari beberapa baris, tergantung berapa jenis barang yang dibeli.

4.1.2 Menghapus Kantong Plastik

Pada saat membaca *notepad*, kantong plastik juga dihitung sebagai barang yang dibeli konsumen. Olehkarena itu tahap kedua yang dilakukan adalah mengeluarkan item kantong plastik dari transaksi. Sebab pada saat analisis item kantong plastik tidak dianggap sebagai item yang dibeli konsumen.

Data yang akan digunakan untuk analisis dibagi menjadi dua yaitu data asli dengan nama barang detail yang dinamai dengan data barang detail dan yang kedua adalah data asli yang kemudian diolah sehingga didapatkan data barang dengan nama yang lebih umum yang dinamai dengan data barang umum.

4.1.3 Membuat Kode Barang

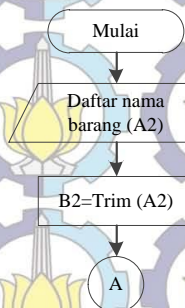
Tahap ketiga yang dilakukan dalam *preprocessing* adalah membuat kode barang. Pada tahap ini terdapat perbedaan pembuatan kode barang pada data dengan nama barang detail dan nama barang yang telah digeneralkan.

Pada hasil *preprocessing* tahap pertama nama barang yang dibeli konsumen tertulis berkali-kali, tergantung berapa kali barang tersebut dibeli dalam keseluruhan transaksi. Olehkarena itu dicari *unique* dari daftar barang yang dibeli oleh konsumen dengan menggunakan *package* dplyr dan ddplyr. *Syntax* yang digunakan dalam mencari daftar barang ini tertulis dalam lampiran 4. *Output* yang dihasilkan pada tahap ini adalah kolom

pertama yang berisi angka 1 sampai dengan sejumlah nama barang *unique*. Kolom kedua adalah berisi daftar barang. Satu barang hanya ditulis sebanyak satu kali. Kolom terakhir berisi transaksi berapa sajakah yang mengandung nama barang tersebut.

Pada saat membuat nama barang umum, peneliti tidak bisa menggunakan kode *PLU (Pice Look Up-unit)* pada barang sebagai dasar dalam menggeneralkan nama barang. Sebab pada kode plu dari setiap barang tatanan kode yang diberikan tidak teratur. Tidak diketahui secara pasti pada angka keberapakah yang membedakan ukuran dari barang-barang yang dibeli konsumen. Olehkarena itu dalam pengeneralan nama barang, peneliti menggunakan cara menghapus keterangan mili yang ada pada kata paling belakang nama barang.

Pada data transaksi yang telah dibaca pada tahap pertama didapatkan bahwa penulisan nama barang tidak konsisten. Misalnya adalah pada beberapa barang air mineral dituliskan secara lengkap sebagai air mineral. Namun pada beberapa barang yang lain ada yang menggunakan singkatan. Sehingga kata air mineral berubah menjadi air mnral. Jadi pada tahap mengeneralkan nama barang, daftar barang yang digunakan sebagai dasar adalah daftar barang yang dijual yang diperoleh dari minimarket *Kl mart*. Pada daftar barang yang dijual, nama barang dituliskan secara konsisten sehingga pada tahap pengeneralan nama barang bisa menggunakan data daftar barang penjualan tersebut.



Gambar 4.2 Flowchart Membuat Daftar Barang Umum



Gambar 4.2 Flowchart Membuat Daftar Barang Umum (Lanjutan)

Setelah didapatkan daftar nama barang secara general lalu melakukan pengkodean nama barang secara umum dengan cara

yang sama pada saat melakukan pengkodean pada daftar barang detail yaitu menggunakan *syntax* pada lampiran 4. Selanjutnya membuat daftar kode plu barang dan kode barang umum baru dengan menggunakan fungsi *vlookup*.

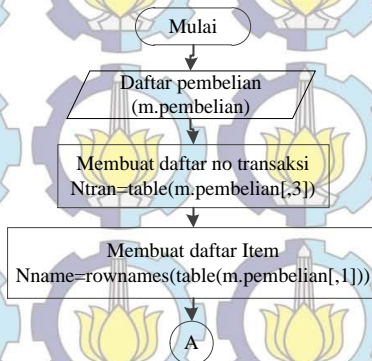
Pada tahap ke tiga telah dilakukan pembuatan kode barang baru baik untuk barang detail maupun untuk barang umum. Sehingga pada tahap ke tiga didapatkan dua file. Pada data barang detail, kolom pertama berisi nama barang awal dan kolom kedua berisi kode barang baru. Sedangkan untuk data barang umum, kolom pertama berisi kode plu barang dan kolom kedua berisi kode barang umum baru.

4.1.4 Mengganti Nama Barang dengan Kode Barang

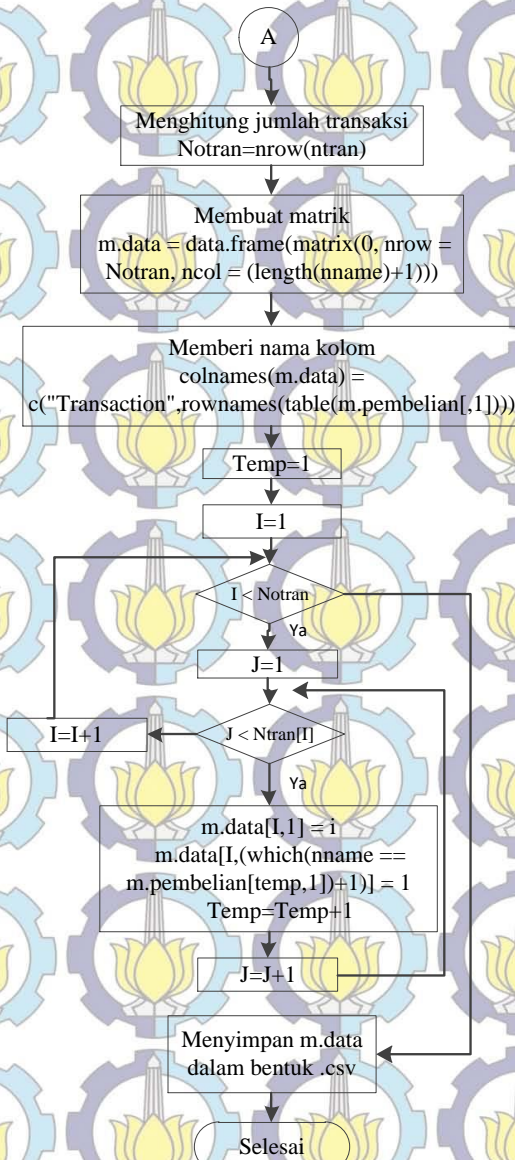
Tahap keempat pada proses *preprocessing* adalah mengganti nama barang dengan kode barang baru. Pada data barang detail, hasil pada tahap kedua, kolom yang berisi nama barang diganti dengan kode barang baru dengan menggunakan fungsi *vlookup*. Pada data dengan nama umum juga menggunakan cara yang sama. Namun kode barang baru yang digunakan adalah kode barang baru untuk data umum.

4.1.5 Membuat Struktur Data Biner

Selanjutnya, pada tahap kelima adalah membuat data pada hasil proses *preprocessing* ke empat menjadi data dengan struktur data biner. Berikut ini adalah langkah-langkah dalam membuat struktur data biner.



Gambar 4.3 Flowchart Membuat Struktur Data Biner



Gambar 4.3 Flowchart Membuat Struktur Data Biner (Lanjutan)

4.2 Statistika Deskriptif Persebaran Pembelian Konsumen

Pada analisis yang dilakukan terdapat dua macam barang yang diolah. Yang pertama adalah data detail dan yang kedua adalah data barang dengan nama yang sudah digeneralkan.

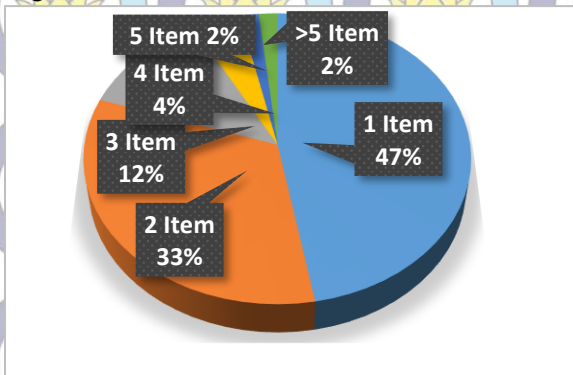
4.2.1 Data Barang Detail

Terdapat 9871 transaksi dan 1286 item yang digunakan dalam pada data barang detail. Berikut ini adalah statistika deskriptif untuk jumlah item yang dibeli pada barang detail.

Tabel 4.1 Statistika Deskriptif Jumlah Item Data Barang Detail dan Barang Umum

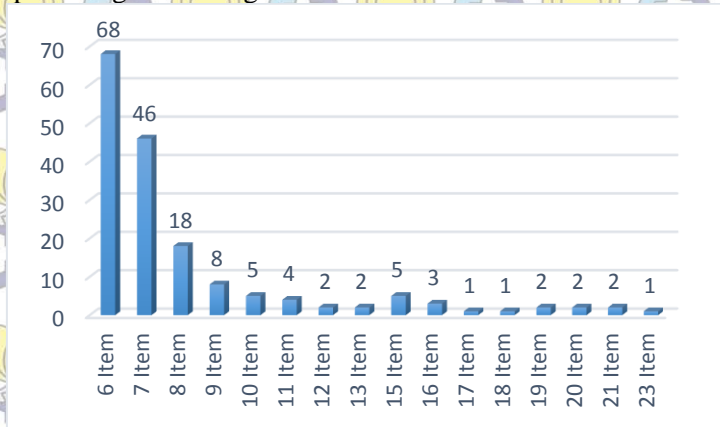
Data Barang	Minimum	Median	Rata-rata	Modus	Maksimum
Umum	1	2	1,893	1	23
Detail	1	2	1,889	1	23

Berdasarkan Tabel 4.1 di atas dapat diketahui bahwa pada transaksi yang dilakukan konsumen *Kl mart*, jumlah item paling sedikit yang dibeli adalah 1 item dan paling banyak adalah 23 item. Jumlah item yang paling banyak dibeli adalah 1 item barang. Median dari jumlah item yang dibeli dalam transaksi adalah 2 item barang. Dari keseluruhan transaksi, rata-rata jumlah item yang dibeli konsumen adalah 2.



Gambar 4.4 Persebaran Jumlah Item yang Dibeli pada Data Barang Detail

Berdasarkan Gambar *pie chart* di atas, terlihat bahwa persebaran pembelian item di minimarket *KImart* tidak merata. Masih jarang konsumen yang melakukan pembelian untuk keperluan bulanan di minimarket ini. Hal ini ditunjukkan dengan jumlah pembelian enam atau lebih item masih sangat sedikit yaitu hanya 2% dari total transaksi saja. Sedangkan, 47% dari transaksi yang dilakukan konsumen hanya membeli sejumlah 1 item saja. 33% yang lainnya adalah transaksi yang dilakukan pada pembelian 2 item. Lalu pada transaksi pembelian 3 item, 4 item, 5 item, dan yang lebih dari 5 item masing-masing hanya sebesar 12%, 4%, 2% dan 2% saja. Gambaran secara detail tentang jumlah pembelian konsumen pada 6 item barang atau lebih dapat dilihat pada diagram batang di bawah ini.



Gambar 4.5 Diagram Batang Pembelian pada Data Barang Detail

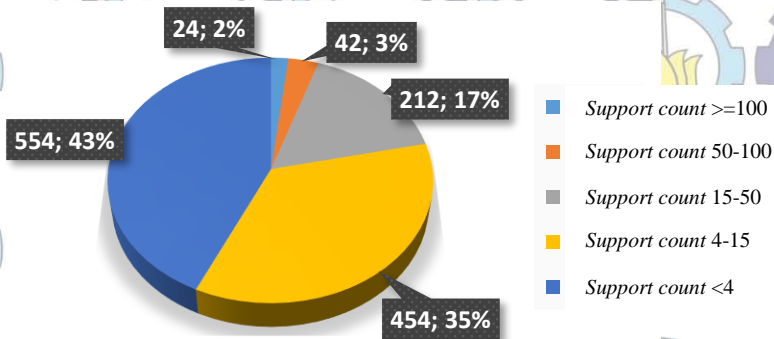
Seperti yang telah disebutkan sebelumnya, terdapat 9871 transaksi yang telah dilakukan di minimarket *KImart*. 2% diantaranya adalah transaksi yang dilakukan untuk membeli enam item atau lebih. Berdasarkan gambaran diagram batang diatas dapat diketahui bahwa pada pembelian 6 item, 7 item, dan 8 item jumlah pembelian yang dilakukan, masing-masing hanya sebesar 68 transaksi, 46 transaksi, dan 18 transaksi saja. Jumlah ini tergolong sedikit, sebab jumlah keseluruhan transaksi yang

dilakukan adalah sebanyak 9871 transaksi. Pada pembelian sembilan item atau lebih bahkan hanya terdapat kurang dari sepuluh transaksi saja. Pada pembelian 9 item, hanya terdapat pada 8 transaksi. Selanjutnya adalah pada pembelian 10 item dan 15 item hanya dibeli dalam 5 transaksi. Sisanya yaitu pada pembelian 11 item dilakukan dalam 4 transaksi, 16 item pada 3 transaksi. Pada pembelian 12,13,19,20, dan 21 item hanya dilakukan pada 2 transaksi. Jumlah transaksi yang paling sedikit adalah satu yaitu pada pembelian 17,18, dan 23 item saja. Berikut ini adalah statistika deskriptif dari *support count* item yang dibeli konsumen.

Tabel 4.2 Statistika Deskriptif *Support Count* Data Barang Detail dan Barang Umum

Data Barang	Minimum	Median	Rata-rata	Modus	Maksimum
Detail	1	4	14,533	1	953
Umum	1	5	15,7766	1	1499

Berdasarkan Tabel dapat diketahui bahwa dari 1286 item yang dibeli konsumen, *support count* paling sedikit adalah 1. Jadi masih ada item yang dibeli hanya satu kali dari 9871 transaksi yang telah dilakukan. Sedangkan nilai maksimum *support count* pada data barang detail adalah 953. Artinya dari 1286 item yang dibeli konsumen, terdapat item yang dibeli sebanyak 953 kali. Nilai median dari jumlah pembelian dari setiap item adalah 4. Sedangkan rata-rata jumlah pembelian dari setiap itemnya adalah 15 kali. Namun jika dilihat dari nilai modulusnya, kebanyakan item yang dibeli di minimarket *KI mart* adalah sejumlah 1 kali. Berikut ini adalah persebaran jumlah pembelian dari setiap item data barang detail pada minimarket *KI mart*.



Gambar 4.6 Persebaran *Support Count* pada Data Barang Detail

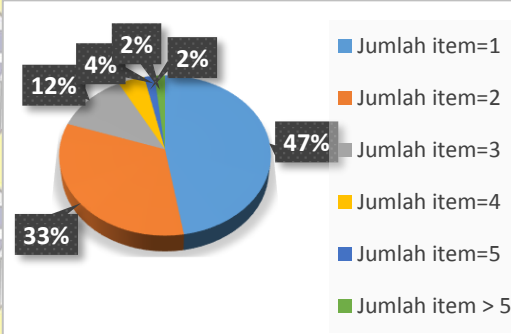
Gambar di atas menunjukkan persebaran *support count* dari 1286 item yang dibeli konsumen *K1mart*. Berdasarkan Gambar 4.6 terlihat bahwa hanya terdapat sedikit item yang dibeli lebih dari 100 kali, 43% dari total item yang dibeli konsumen hanya dibeli kurang dari 4 kali. Jumlah item yang dibeli antara 4 sampai 15 kali hampir sama banyaknya yaitu 35% dari total item yaitu ada sejumlah 454 item. Lalu 17% dari total item pada minimarket *K1mart* dibeli antara 15 sampai dengan 50 kali. Sedangkan untuk jumlah pembelian antara 50 sampai dengan 100 kali hanya terdapat 3% dari total item yang ada saja dan yang paling sedikit adalah item yang dibeli lebih dari 100 kali. Dari total item yaitu sebanyak 1286 hanya terdapat 24 item saja yang telah dibeli lebih dari 100 kali.

4.2.2 Data Barang Umum

Setelah mengetahui karakteristik dari data transaksi dengan nama barang detail, maka selanjutnya dilakukan analisis terhadap karakteristik dari data transaksi dengan nama barang lebih general. Statistika deskriptif untuk jumlah item yang dibeli konsumen ditunjukkan pada Tabel 4.1.

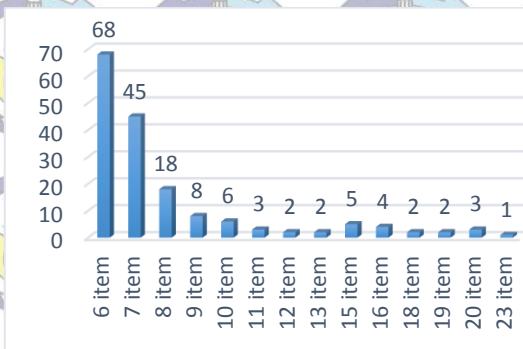
Tabel 4.1 menunjukkan karakteristik dari jumlah item yang dibeli konsumen *K1mart*. Konsumen *K1mart* membeli paling sedikit 1 item dan paling banyak adalah sejumlah 44 item. Jumlah item yang paling sering dibeli oleh konsumen adalah 1 item

barang, sedangkan nilai mediannya adalah 2 item dan nilai rata-ratanya adalah 2 item. Berikut ini adalah diagram batang untuk jumlah item yang dibeli konsumen.



Gambar 4.7 Persebaran Jumlah Item pada Data Barang Umum

Berdasarkan Gambar di atas dapat dilihat bahwa persebaran jumlah item yang paling banyak dibeli konsumen adalah 1 item saja yaitu sebanyak 47%. Selanjutnya adalah pembelian untuk 2 item yaitu sebanyak 33% dan sisanya membeli lebih dari 2 item. Berikut ini adalah diagram batang untuk jumlah item yang dibeli mulai dari 9 item sampai dengan 44 item.



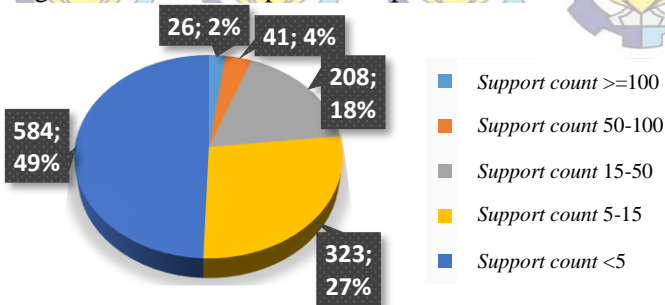
Gambar 4.8 Diagram Batang Jumlah Item 6 Sampai dengan 23

Dari diagram batang di atas, dapat disimpulkan bahwa jumlah item yang dibeli berbanding terbalik dengan jumlah

transaksi yang dilakukan. Kebanyakan konsumen melakukan transaksi di *Klmart* hanya membeli sedikit barang saja. Masih sedikit konsumen yang membeli belanja bulanan di *Klmart*.

Setelah data barang dengan nama detail diumumkan, jumlah macam-macam jenis item yang dibeli konsumen menjadi lebih sedikit. Pada data transaksi dengan nama barang detail terdapat 1286 jenis item yang dibeli. Sedangkan pada data transaksi dengan nama barang umum terdapat 1182 jenis item yang dibeli. Berikut ini adalah statistika deskriptif untuk nilai *support count* item pada data barang umum.

Minimal jumlah transaksi yang telah dilakukan dalam membeli barang dengan kode nama umum di minimarket *Klmart* adalah sebanyak satu transaksi dan jumlah tertinggi transaksi dari suatu item adalah 1499 kali. Berdasarkan Tabel 4.2 terlihat bahwa jumlah maksimum, median dan nilai rata-rata pembeli per item pada data barang dengan nama umum mengalami peningkatan jika dibandingkan dengan data barang dengan nama detail. Pada jumlah maksimum pembelian item awalnya adalah 953 kali, selanjutnya pada data barang umum meningkat menjadi 1499 kali. Namun pada jumlah nilai modus untuk jumlah pembelian item masih tetap sama yaitu satu. Jadi masih terdapat banyak item yang dibeli hanya satu kali saja. Untuk mengetahui persebaran jumlah pembelian konsumen untuk semua item pada data barang dengan nama umum dapat dilihat pada Gambar 4.9 di bawah ini.



Gambar 4.9 Persebaran *Support Count* yang Dibeli pada Data Barang Umum

Persebaran jumlah pembelian item yang paling tinggi terdapat pada pembelian dengan jumlah kurang dari lima item. Terdapat 49% atau sebesar 584 item yang dibeli kurang dari kali. Sedangkan nilai paling rendah yaitu pada pembelian lebih dari atau sama dengan 100 kali. Hanya terdapat 26 item atau 2% dari total item saja yang pernah yang dibeli lebih dari atau sama dengan 100 kali.

4.3 Pola Perilaku Konsumen dengan Algoritma *FP-Growth*

Pada penelitian ini jumlah transaksi yang digunakan dalam analisis adalah 9871 transaksi baik untuk data dengan nama detail maupun pada data dengan nama barang umum. Terdapat 1286 jenis barang pada data dengan nama detail. Sedangkan pada data dengan nama umum hanya terdapat 1182 jenis barang saja.

4.3.1 Data Barang Detail

Minimum *support* yang digunakan adalah 0,0015% atau minimal item telah dibeli sebanyak 15 kali. Berdasarkan nilai *support* tersebut, didapatkan 278 item yang *frequent* untuk 1 *itemset* dan 15 item yang *frequent* untuk 2 *itemset*. Berikut ini adalah 5 item dengan nilai *support* tertinggi untuk 1 temset.

Tabel 4.3 1 *Itemset* dengan *Support* Tertinggi pada Data Barang Detail Algoritma *FP-Growth*

No	Item	Keterangan Item	Support
1	589	1062730 Club Air Mnral 600ml	0,097
2	80	0012700 Aqua Air Mineral 600	0,067
3	590	1062740 Club Air Mnrl 1500ml	0,058
4	79	0012690 Aqua Air Mineral1500	0,041
5	1045	1321530 Ades Air Mineral 600	0,031

Berdasarkan Tabel 4.3 di atas dapat diketahui bahwa 5 item yang paling sering dibeli pelanggan adalah item dengan kode 589, 80, 590, 79, dan 1045. Sebanyak 9,7% dari transaksi yang dilakukan konsumen mengandung item 589. Artinya dari 9871 transaksi, terdapat 958 transaksi yang dilakukan untuk membeli item 589. Item ke dua yang paling sering dibeli pelanggan adalah item dengan kode 80. Dari keseluruhan transaksi yang dilakukan 6,7% diantaranya mengandung item dengan kode 80. Artinya dari

9871 transaksi terdapat 662 transaksi yang mengandung item 80. Pada item ke tiga yaitu item dengan kode 590, item ini dibeli dalam 5,8% transaksi yaitu sejumlah 573 transaksi. Item dengan kode barang 79 adalah item keempat yang paling sering dibeli pelanggan. Terdapat 4,1% dari transaksi yang dilakukan mengandung item 79. Artinya dari 9871 transaksi yang dilakukan, 405 diantaranya mengandung item 79. Sedangkan untuk item ke lima yang paling sering dibeli pelanggan yaitu item 1045, terdapat 3,1% transaksi yang mengandung item 1045. Berikut ini adalah 5 item dengan nilai *support* tertinggi untuk 2 *itemset*.

Tabel 4.4 2 *Itemset* dengan *Support* Tertinggi pada Data Barang Detail Algoritma *FP-Growth*

No	Item 1	Keterangan Item 1	Item 2	Keterangan Item 2	Support
1	589	1062730 Club Air Mnral 600ml	590	1062740 Club Air Mnrl 1500ml	0,003
2	589	1062730 Club Air Mnral 600ml	937	1235680 Pucuk/H Teh Mlati350	0,003
3	123	0036830 Ultra Uht Choco 250m	452	0509440 Mr.Bread Roti Cokl60	0,003
4	589	1062730 Club Air Mnral 600ml	80	0012700 Aqua Air Mineral 600	0,002
5	589	1062730 Club Air Mnral 600ml	123	0036830 Ultra Uht Choco 250m	0,002

Tabel 4.4 menunjukkan 5 pasang item dengan nilai *support* tertinggi untuk dua *itemset*. Dalam pembelian dua item, terdapat lima pasang item yang sering dibeli pelanggan yaitu item 589 dan item 590, item 589 dan item 937, item 123 dan item 452, item 589 dan item 80, serta item 589 dan item 123. Terdapat 0,3% transaksi yang memuat pembelian antara item 589 dan item 590. Sama halnya dengan item 589 dan item 937 serta item 123 dan item 452, terdapat 0,3% dari transaksi mengandung kedua item

tersebut. Artinya dari 9871 transaksi, terdapat 30 transaksi yang mengandung item 589 dan 590, item 589 dan 937, serta item 123 dan 452. Item 589, dan item 80 serta item 589 dan item 123 juga memiliki support yang sama yaitu 0,002. Artinya 0,2% dari transaksi yang dilakukan mengandung pasangan item tersebut yaitu sebanyak 20 dari 9871 transaksi. Berdasarkan *frequent itemset* yang telah didapatkan, terdapat 9 *rule* yang dihasilkan. *Rule-rule* tersebut dapat dilihat dalam Tabel 4.7 berikut ini.

Tabel 4.5 2 Rule pada Data Barang Detail Algoritma FP-Growth

No	LHS	RHS	Support	Confidence	Lift
1	937 1235680 Pucuk/H Teh Mlati350	589 1062730 Club Air Mnral 600ml	0,0028	0,1279	1,3243
2	420 0454160 Ultra Teh Kotak 200	80 0012700 Aqua Air Mineral 600	0,0015	0,1293	1,9399
3	452 0509440 Mr.Bread Roti Cokl60	589 1062730 Club Air Mnral 600ml	0,0024	0,1356	1,4044
4	452 0509440 Mr.Bread Roti Cokl60	123 0036830 Ultra Uht Choco 250m	0,0025	0,1412	4,6629
5	64 0008040 Beng-Beng Choco 22g	589 1062730 Club Air Mnral 600ml	0,0016	0,1495	1,5488
6	168 0060410 Indomi Goreng Spc 80	84 0013550 Indomi Grg Pedas 80g	0,0016	0,1818	47,2297
7	753 1153830 Soba Mi Stik S.Bld24	752 1153820 Soba Mi Stik A.Bkr24	0,0016	0,2909	87,0171

Tabel 4.5 2 Rule pada Data Barang Detail Algoritma FP-Growth
(Lanjutan)

No	LHS	RHS	Support	Confidence	Lift
8	84 0013550 Indomi Grg Pedas 80g	168 0060410 Indomi Goreng Spc 80	0,0016	0,4211	47,2297
9	752 1153820 Soba Mi Stik A.Bkr24	753 1153830 Soba Mi Stik S.Bld24	0,0016	0,4848	87,071

LHS menunjukkan item pertama yang dibeli konsumen. RHS menunjukkan item yang dibeli selanjutnya oleh konsumen. Dari 9 rule yang dihasilkan pada Tabel 4.5, nilai *confidence* tertinggi yang dihasilkan adalah 48,48% yaitu pada rule ke 9. Sedangkan untuk nilai lift tertinggi yang dihasilkan adalah 87,071 yang terdapat pada rule ke 9 juga. Semua rule yang telah dihasilkan memiliki nilai lift lebih dari satu, sehingga dapat dikatakan bahwa terdapat hubungan antara pembelian item LHS dan item pada RHS pada semua rule yang dihasilkan. Jadi semua rule yang dihasilkan layak untuk dianalisis. Pada rule 1, 0,28% dari keseluruhan transaksi yang dilakukan, item 937 dan 589 dibeli secara bersama-sama dan dalam 12,79% dari pembelian item 937 juga mengandung item 589. Artinya tingkat kepercayaan seseorang yang membeli item 937 juga akan membeli item 589 adalah 12,79%. Nilai lift yang dihasilkan adalah 1,3243. Karena nilai ini lebih dari satu maka dapat dikatakan bahwa pembelian item 937 dan item 589 saling berhubungan.

Pada rule 2. Item 420 dan item 80 dibeli bersama-sama dalam 0,15% dari keseluruhan transaksi. Jika pelanggan membeli item 420 maka pelanggan tersebut juga akan membeli item 80, tingkat kepercayaannya adalah 12,93%. Nilai lift yang dihasilkan adalah 1,9399. Artinya terdapat hubungan antara item 420 dan item 80.

Pada *rule* 3, 0,24% dari seluruh transaksi yang dilakukan item 452 dan item 589 dibeli secara bersama-sama. 13,56% pelanggan yang membeli item 452 juga membeli 589. Pada *rule* 4, item 452 dan item 123 dibeli secara bersama-sama dalam 0,25% dari total transaksi atau sebanyak 25 kali dan 14,12% dari transaksi yang dilakukan untuk membeli item 452, juga mengandung item 123. *Rule* yang kelima yaitu jika konsumen membeli item 64 juga akan membeli item 589. Dimana terdapat 0,16% dari total transaksi yang dilakukan, konsumen membeli item 64 dan 589 secara bersama-sama. 14,95% konsumen yang membeli item 64 juga akan membeli item 589.

Rule ke enam yang ditemukan yaitu jika konsumen membeli item 168 maka konsumen tersebut juga akan membeli item 84. Dari keseluruhan transaksi yang dilakukan konsumen K1mart, terdapat 0,16% diantaranya yang membeli item 168 dan item 84 secara bersama-sama dan 18,18% konsumen yang membeli item 168 juga membeli item 84. Pada *rule* 7, *association rule* yang ditemukan adalah jika konsumen membeli item 753 juga akan membeli item 752. Terdapat 0,16% dari total transaksi yang telah membeli item 753 dan item 752 secara bersama-sama dan 29,09% konsumen yang membeli item 753 juga membeli item 752.

Pada *rule* ke delapan didapatkan pola hubungan bahwa jika konsumen membeli item 84 maka pelanggan juga akan membeli item 168. Terdapat 0,16% dari total transaksi yang dilakukan konsumen, dilakukan untuk membeli item 84 dan item 168 secara bersama-sama. dari semua transaksi yang dilakukan untuk membeli item 84, terdapat 42,11% diantaranya yang dilakukan untuk membeli item 168 juga.

Rule yang memiliki nilai *confidence* paling tinggi adalah *rule* ke sembilan. Pola hubungan pada *rule* ke sembilan yaitu jika konsumen membeli item 752 maka konsumen juga akan membeli item 753. Item 752 dan item 753 dibeli secara bersama-sama dalam 0,16% dari total transaksi. Berdasarkan nilai *confidence*

yang didapatkan, dapat dikatakan bahwa 48,48% konsumen yang membeli item 752 juga akan membeli item 753.

4.3.2 Data Barang Umum

Selanjutnya akan dilakukan analisis pada data transaksi dengan nama barang umum. Minimal *support* yang digunakan pada data transaksi dengan nama umum adalah 0,0015. Artinya item yang dibeli lebih dari atau sama dengan lima belas layak untuk diperhitungkan. Pada data barang yang telah digeneralkan, dengan menggunakan minimal *support* 0,0015 terdapat 298 item yang *frequent*. Item yang *frequent* ini terdiri dari 275 untuk 1 *itemset*, 23 untuk 2 *itemset*, dan tidak terdapat item yang *frequent* pada 3 *itemset*. Berikut ini adalah lima item dengan *support* tertinggi pada 1 *itemset*.

Tabel 4.6 1 *Itemset* dengan *Support* Tertinggi pada Data Barang Umum
Algoritma *FP-Growth*

No	Item	Keterangan Item	Support
1	256	Club Air Mineral Btl	0,152
2	72	Aqua Air Mineral Prima Btl	0,067
3	71	Aqua Air Mineral Btl	0,041
4	557	Gudang Garam Rokok Filter Surya (Pcs) Bks	0,031
5	39	Ades Air Mineral Btl	0,031

Lima item yang paling sering dibeli pada minimarket K1mart adalah item 256, 72, 71, 557, dan 39. Item 256 telah dibeli dalam 15,2% dari keseluruhan transaksi yaitu sebanyak 1501 kali. Pada item 72, item ini telah dibeli dari 6,7% dari total transaksi yaitu sebanyak 6614 kali. Item 71 dibeli pada 4,1% dari transaksi atau sebanyak 405 kali. Sedangkan item 557 dan item 39 memiliki nilai *support* yang sama yaitu 0,031. Artinya item 557 dan item 39 masing-masing dibeli dalam jumlah transaksi yang sama yaitu 3,1% dari total transaksi atau sebanyak 306 kali.

Maksimum *itemset* yang *frequent* adalah 2 *itemset*. Sebab tidak ada pembelian pada 3 *itemset* atau lebih yang memenuhi kriteria minimum *support*. Pada 2 *itemset* terdapat 23 item yang *frequent*. Berikut ini adalah 5 item dengan *support* tertinggi pada barang yang telah digeneralkan

Tabel 4.7 2 *Itemset* dengan *Support* Tertinggi pada Data Barang Umum
Algoritma *FP-Growth*

No	Item 1	Keterangan Item 1	Item 2	Keterangan Item 2	Support
1	256	Club Air Mineral Btl	1299	Pucuk Harum Minuman Teh Melati Btl	0,004
2	256	Club Air Mineral Btl	72	Aqua Air Mineral Prima Btl	0,003
3	256	Club Air Mineral Btl	71	Aqua Air Mineral Btl	0,003
4	256	Club Air Mineral Btl	1646	Ultra Susu Uht Steril Slim Chocolate Tpk	0,003
5	256	Club Air Mineral Btl	1255	Pocari Sweat Minuman Isotonik Btl	0,003

Nilai *support* tertinggi pada dua *itemset* adalah 0,004. Artinya paling banyak terdapat 40 pembelian barang dengan dua item yang sama. Pasangan item-item yang *frequent* tersebut antara lain item 256 dan 1299, item 256 dan 72, item 256 dan 71, item 256 dan 1646, item 256 dan item 1255. Pada item 256 dan 1299, item ini dibeli secara bersama-sama dalam 0,4% dari total pembelian yaitu sebanyak 40 kali. Sedangkan keempat pasang item yang lainnya hanya dibeli secara bersama-sama dalam 30 transaksi saja atau sebesar 0,3% dari total transaksi.

Setelah didapatkan item yang *frequent*, tahap selanjutnya adalah membentuk *rule*. Dengan menggunakan minimum *support* sebesar 0,15% dan minimum *confidence* sebesar 5%, terdapat 17 *rule* yang dihasilkan. Dari 17 *rule* yang dihasilkan hanya terdapat 11 *rule* saja yang memiliki nilai *lift* lebih dari 1.

Tabel 4.8 Rule pada Data Barang Umum Algoritma *FP-Growth*

No	LHS	RHS	Support	Confidence	Lift
1	1299 Pucuk Harum Minuman Teh Melati Btl	256 Club Air Mineral Btl	0,0035	0,1598	1,0524
2	1057 Mr.Bread Roti Manis Isi Coklat Special Pck	256 Club Air Mineral Btl	0,0031	0,1751	1,1533
3	1057 Mr.Bread Roti Manis Isi Coklat Special Pck	1646 Ultra Susu Uht Steril Slim Chocolate Tpk	0,0025	0,1412	4,6629
4	103 Bear Brand Susu Encer Steril Klg	256 Club Air Mineral Btl	0,0022	0,1618	1,0652
5	224 Chupa Chups Candy Lollipops Assorted Pcs	256 Club Air Mineral Btl	0,0020	0,2083	1,3719
6	112 Beng-Beng Wafer Chocolate Pck	256 Club Air Mineral Btl	0,0017	0,1589	1,0462
7	637 Indomie Mie Goreng Plus Special Pck	636 Indomie Mie Goreng Pedas Pck	0,0016	0,1818	47,2297

Tabel 4.8 Rule pada Data Barang Umum Algoritma *FP-Growth* (Lanjutan)

No	LHS	RHS	Support	Confidence	Lift
8	1480 Soba Mie Stick Ala Jepang Premium Sambal Balado Pck	1479 Soba Mie Stick Ala Jepang Premium Ayam Bakar Pck	0,0016	0,2909	87,0171
9	636 Indomie Mie Goreng Pedas Pck	637 Indomie Mie Goreng Plus Special Pck	0,0016	0,4211	47,2297
10	1479 Soba Mie Stick Ala Jepang Premium Ayam Bakar Pck	1480 Soba Mie Stick Ala Jepang Premium Sambal Balado Pck	0,0016	0,4848	87,0171
11	1651 Soba Mie Stick Ala Jepang Premium Ayam Bakar Pck	72 Soba Mie Stick Ala Jepang Premium Sambal Balado Pck	0,0015	0,1293	1,9398

Hubungan antar LHS dan RHS yang dihasilkan dikatakan signifikan apabila memiliki nilai lift lebih dari satu. *Rule* dengan nilai lift lebih besar menandakan adanya hubungan antar LHS dan RHS yang lebih signifikan. Pada data barang umum, terdapat 17 *rule* yang menarik namun hanya ada 11 *rule* saja yang memiliki hubungan antara LHS dan RHS yang signifikan.

Nilai *support* menunjukkan seberapa sering item dalam LHS dan item dalam RHS dibeli secara bersama-sama dalam transaksi. Sedangkan nilai *confidence* menunjukkan seberapa

besar keyakinan konsumen yang membeli item pada LHS juga akan membeli item pada RHS. *Rule* yang memiliki nilai *confidence* tertinggi terdapat pada *rule* ke 10 yaitu sebesar 48,48%.

Rule pertama yang dihasilkan yaitu jika konsumen membeli item 1299 maka konsumen tersebut juga akan membeli item 256. 0,35% transaksi yang dilakukan konsumen untuk membeli item 1299 dan item 256 secara bersama-sama. terdapat 15,98% transaksi yang dilakukan konsumen untuk membeli item 1299 juga terdapat item 256 yang juga dibeli dalam transaksi.

Pada *rule* kedua, jika konsumen membeli item 1057 maka konsumen juga akan membeli item 256. 0,31% dari transaksi yang dianalisis, pelanggan membeli item 1057 dan item 256 secara bersama-sama dan 17,51% konsumen yang membeli item 1057 juga akan membeli item 256.

Rule ketiga adalah jika konsumen membeli item 1057 maka konsumen tersebut juga akan membeli item 1646. Terdapat 0,25% transaksi yang dilakukan konsumen untuk membeli item 1057 dan item 1646 secara bersama-sama. Tingkat kepercayaan konsumen yang membeli item 1057 juga akan membeli item 1646 adalah sebesar 14,12%.

Rule keempat yang didapatkan adalah jika konsumen membeli item 103 maka konsumen juga akan membeli item 256. 0,22% dari transaksi yang telah dilakukan, konsumen membeli item 103 dan 256 secara bersama-sama. Dari semua transaksi yang dilakukan untuk membeli item 103, 16,18% diantaranya dilakukan untuk membeli item 256 juga.

Pola pembelian yang kelima yaitu jika konsumen membeli item 224 maka akan membeli item 256. Terdapat 0,16% konsumen yang telah membeli item 224 dan item 256 secara bersama-sama. tingkat kepercayaan pada konsumen yang membeli item 224 juga akan membeli item 256 adalah sebesar 14,95%.

Pola pembelian konsumen yang keenam adalah jika konsumen membeli item 112 maka akan membeli item 256.

0,16% dari total transaksi, konsumen membeli item 112 dan item 256 secara bersama-sama. 18,18% pelanggan yang membeli item 112 juga akan membeli item 256.

Rule yang ketujuh yang dihasilkan adalah jika konsumen membeli item 637 maka konsumen juga akan membeli item 636. Terdapat 0,16% transaksi yang telah dilakukan konsumen untuk membeli item 637 dan item 636 secara bersama-sama. tingkat kepercayaan bahwa konsumen yang membeli item 637 juga akan membeli item 636 adalah sebesar 29,09%.

Rule kedelapan yaitu jika konsumen membeli item 1480 maka konsumen juga akan membeli item 1479. 0,16% dari transaksi yang dilakukan konsumen, item 1480 dan item 1479 dibeli secara bersama-sama. seorang konsumen yang membeli item 1480, 29,09% diantaranya juga akan membeli item 1479.

Pada *rule* ke sembilan, jika konsumen membeli item 636 maka konsumen tersebut juga akan membeli item 637. Item 636 dan 637 dibeli secara bersama-sama dalam 0,16% dari total transaksi yang dilakukan konsumen. 42,11% konsumen yang membeli item 636 juga membeli item 637.

Rule ke sepuluh yang dihasilkan adalah jika konsumen membeli item 1479 maka akan membeli item 1480. 0,16% dari transaksi dilakukan konsumen untuk membeli item 1479 dan item 1480 secara bersama-sama. Tingkat kepercayaan konsumen yang membeli item 1479 juga akan membeli item 1480 adalah sebesar 48,48%. Ini adalah tingkat kepercayaan tertinggi pada *rule* yang dihasilkan untuk data transaksi dengan nama barang umum.

Rule terakhir yang didapatkan yaitu jika konsumen membeli item 1651 maka konsumen juga akan membeli item 72. Terdapat 0,15% transaksi yang dilakukan konsumen untuk membeli item 1651 dan item 72 secara bersama-sama. 12,93% konsumen yang membeli item 1651 juga akan membeli item 72.

4.4 Pola Perilaku Konsumen dengan Algoritma Apriori

Algoritma ke dua yang digunakan dalam analisis pola pembelian konsumen pada minimarket *K1mart* adalah apriori. Sama seperti pada algoritma *FP-Growth*, minimal *support* yang digunakan adalah 0,15%.

4.4.1 Data Barang Detail

Pada algoritma apriori, untuk data barang detail, jumlah *frequent itemset* yang dihasilkan sama yaitu terdapat 278 pada 1 *itemset* dan 15 pada 2 *itemset*. Berikut ini adalah 5 hasil item yang *frequent* untuk 1 *itemset* dengan nilai *support* tertinggi.

Tabel 4.9 1 *Itemset* dengan *Support* Tertinggi pada Data Barang Detail Algoritma Apriori

No	Item	Keterangan Item	Support
1	589	1062730 Club Air Mnral 600ml	0,0965
2	80	0012700 Aqua Air Mineral 600	0,0667
3	590	1062740 Club Air Mnrl 1500ml	0,0582
4	79	0012690 Aqua Air Mineral1500	0,0411
5	1045	1321530 Ades Air Mineral 600	0,0306

Pada data barang detail, nilai *support* tertinggi yang dihasilkan adalah 9,65% atau sebanyak 953 kali pembelian. Pada 1 *itemset*, barang yang paling sering dibeli adalah item 589. Pada item 80 memiliki nilai *support* 0,0667. Artinya item 80 dibeli pada 6,67% dari total transaksi yaitu sebanyak 659 transaksi. Item 590 dibeli dalam 5,82% dari total transaksi atau sebanyak 575 kali. Item 79 dibeli dalam 4,11% dari keseluruhan transaksi yang dilakukan yaitu sebanyak 406 kali. Item kelima yang paling sering dibeli konsumen adalah item 1045. Item 1045 telah dibeli sebanyak 3,06% dari total transaksi yang telah dilakukan atau sebanyak 303 kali.

Pada 2 *itemset* terdapat 15 item yang *frequent*. Berikut ini adalah 5 item yang *frequent* dengan *support* tertinggi pada data barang detail dengan 2 *itemset*.

Tabel 4.10 2 *Itemset dengan Support Tertinggi pada Data Barang Detail Algoritma Apriori*

No	Item 1	Keterangan Item 1	Item 2	Keterangan Item 2	Support
1	589	1062730 Club Air Mnral 600ml	590	1062740 Club Air Mnrl 1500ml	0,0029
2	589	1062730 Club Air Mnral 600ml	937	1235680 Pucuk/H Teh Mlati350	0,0028
3	123	0036830 Ultra Uht Choco 250m	452	0509440 Mr.Bread Roti Cokl60	0,0025
4	452	0509440 Mr.Bread Roti Cokl60	589	1062730 Club Air Mnral 600ml	0,0024
5	123	0036830 Ultra Uht Choco 250m	80	0012700 Aqua Air Mineral 600	0,0022

Pasangan item yang memiliki lima nilai *support* tertinggi adalah item 589 dan 590, item 589 dan 937, item 123 dan 452, item 452 dan 589 serta item 123 dan 80. Item 589 dan 590 memiliki nilai *support* sebesar 0,0029. Artinya item 589 dan 590 telah dibeli secara bersama-sama dalam 0,29% dari seluruh transaksi atau sebanyak 29 kali. Item 589 dan 937 telah dibeli secara bersama-sama dalam 0,28% dari total transaksi yaitu sebanyak 28 kali. Item 123 dan 452 telah dibeli secara bersama-sama dalam 0,25% dari total transaksi yaitu sebanyak 25 kali. Item 452 dan 589 memiliki nilai *support* tidak jauh berbeda dari tiga pasang item sebelumnya yaitu 0,0024. Item 452 dan 589 telah dibeli secara bersama-sama dalam 0,24% dari total atau 24 kali. Lima pasangan item dengan nilai *support* tertinggi yang terakhir adalah item 123 dan item 80 dengan nilai *support* 0,0022. Item ini telah dibeli secara bersama-sama dalam 0,22% dari total transaksi yang dilakukan di minimarket *KI mart* atau sebanyak 22 kali pembelian.

Tidak ada 3 *frequent itemset* yang didapatkan. Total terdapat 293 item yang *frequent*. Berdasarkan algoritma apriori, terdapat 4004 *rule* yang didapatkan pada data barang detail. Berikut ini 10 *rule* dengan nilai support tertinggi.

Tabel 4.11 Rule pada Data Barang Detail Algoritma Apriori

No	LHS	RHS	Support	Confidence	Lift
1	590 1062740 Club Air Mnrl 1500ml	589 1062730 Club Air Mnral 600ml	0,0029	0,0504	0,5224
2	937 1235680 Pucuk/H Teh Mlati350	589 1062730 Club Air Mnral 600ml	0,0028	0,1278	1,3243
3	452 0509440 Mr.Bread Roti Cokl60	123 0036830 Ultra Uht Choco 250m	0,0025	0,1412	4,6629
4	123 0036830 Ultra Uht Choco 250m	452 0509440 Mr.Bread Roti Cokl60	0,0025	0,0836	4,6629
5	452 0509440 Mr.Bread Roti Cokl60	589 1062730 Club Air Mnral 600ml	0,0024	0,1356	1,4044
6	123 0036830 Ultra Uht Choco 250m	80 0012700 Aqua Air Mineral 600	0,0022	0,0736	1,1038
7	123 0036830 Ultra Uht Choco 250m	589 1062730 Club Air Mnral 600ml	0,0022	0,0736	0,7621
8	752 1153820 Soba Mi Stik A.Bkr24	753 1153830 Soba Mi Stik S.Bld24	0,0016	0,4848	87,0171

Tabel 4.11 Rule pada Data Barang Detail Algoritma Apriori (Lanjutan)

No	LHS	RHS	Support	Confidence	Lift
9	753 1153830 Soba Mi Stik S.Bld24	752 1153820 Soba Mi Stik A.Bkr24	0,0016	0,2909	87,0171
10	168 0060410 Indomi Goreng Spc 80	84 0013550 Indomi Grg Pedas 80g	0,0016	0,1818	47,2297

Berdasarkan Tabel 4.11 terdapat 10 pola pembelian dengan nilai *support* tertinggi yang dihasilkan pada algoritma Apriori. Namun dari 10 *rule* yang didapatkan hanya terdapat enam *rule* saja yang memiliki nilai *lift*. Jadi hanya enam *rule* yang memiliki hubungan yang signifikan antara item pada LHS dan item pada RHS nya. *Rule* tersebut antara lain *rule* 2,3,4,5,6,dan *rule* 9.

Pada *rule* dua, pola pembelian konsumen yang dihasilkan adalah jika konsumen membeli item 937 maka akan membeli item 589 juga. Terdapat 0,28% transaksi yang dilakukan konsumen untuk membeli item 937 dan item 589 secara bersama-sama dan 12,78% konsumen yang membeli item 937 juga akan membeli item 589.

Rule ketiga yaitu jika konsumen membeli item 452 maka akan membeli item 123. Item 452 dan item 123 telah dibeli secara bersama-sama dalam 0,25% dari transaksi yang telah dilakukan. Tingkat kepercayaan pada konsumen yang membeli item juga akan membeli item adalah sebesar 14,12%.

Pada *rule* keempat, jika konsumen membeli item 123 maka konsumen juga akan membeli item 452. 0,25% dari total transaksi, item 123 dan item 452 dibeli secara bersama-sama. Dalam semua transaksi yang dilakukan untuk membeli item 123, terdapat 8,36% diantaranya yang juga membeli item 452.

Rule kelima yang dihasilkan adalah jika konsumen membeli item 452 maka konsumen juga akan membeli item 589. Terdapat 0,24% konsumen yang melakukan transaksi untuk

membeli item 452 dan 589 secara bersamaan. 13,56% konsumen yang membeli item 452 juga akan membeli item 589 dalam transaksinya.

Pola pembelian konsumen yang dihasilkan pada *rule* keenam adalah jika konsumen membeli item 123 maka juga akan membeli item 80. Terdapat 0,22% transaksi yang dilakukan oleh konsumen dimana terdapat item 123 dan item 80 yang dibeli dalam transaksi yang sama. Seorang konsumen yang membeli item 123, 7,36% diantaranya juga akan membeli item 80 dalam transaksinya.

Rule ke tujuh yang dihasilkan adalah jika konsumen membeli item 752 maka konsumen juga akan membeli item 753. Dari semua transaksi yang dilakukan konsumen terdapat 0,16% transaksi yang mengandung item 752 dan item 753 secara bersama-sama. 48,48% konsumen yang membeli item 752 juga akan membeli item 753.

Pada *rule* ke sembilan pola pembelian yang dihasilkan adalah jika konsumen membeli item 753 maka konsumen juga akan membeli item 752. Sebanyak 0,16% dari transaksi yang dilakukan konsumen, item 753 dan item 752 dibeli bersama-sama dan 29,09% konsumen yang membeli item 753 juga akan membeli item 752.

Pada *rule* kesepuluh, jika konsumen membeli item 168 maka konsumen juga akan membeli item 84. 0,16% dari total transaksi, item 168 dan item 84 dibeli secara bersama-sama. Dalam semua transaksi yang dilakukan untuk membeli item 168, terdapat 18,18% diantaranya yang juga membeli item 84.

4.4.2 Data Barang Umum

Setelah dilakukan analisis pola pembelian konsumen menggunakan algoritma Apriori pada data dengan nama barang detail, selanjutnya dilakukan analisis pola pembelian konsumen pada data barang dengan nama barang umum. Pada data barang yang telah digeneralkan terdapat 275 item yang *frequent* pada 1 *itemset* yang terbentuk. Berikut ini adalah hasil 5 item dengan nilai *support* tertinggi pada 1 *itemset*.

Tabel 4.12 1 *Itemset dengan Support Tertinggi pada Data Barang Umum Algoritma Apriori*

No	Item	Keterangan Item	Support
1	256	Club Air Mineral Btl	0,1518
2	72	Aqua Air Mineral Prima Btl	0,0667
3	71	Aqua Air Mineral Btl	0,0411
4	557	Gudang Garam Rokok Filter Surya (Pcs) Bks	0,0313
5	39	Ades Air Mineral Btl	0,0306

Pada data barang umum, nilai *support* tertinggi meningkat. Awalnya sebelum digeneralkan nilai *support* tertinggi yang didapatkan adalah 0,0965. Setelah barang lebih digeneralkan nilai *support* tertinggi yang didapatkan yaitu sebesar 0,1518. Berdasarkan tabel terlihat bahwa item yang memiliki nilai *support* tertinggi adalah item 256. Hal ini menunjukkan bahwa item 256 paling banyak dibeli oleh konsumen. Dari 9871 transaksi, terdapat 15,18% transaksi yang membeli item 256. Atau kurang lebih item 256 telah dibeli dalam 1499 transaksi.

Item kedua yang paling sering dibeli konsumen adalah item 72. Item 72 telah dibeli pada 6,67% dari semua transaksi yang telah dilakukan yaitu sekitar 6584 transaksi. Item 71 telah dibeli pada 4,11% transaksi yaitu 406 transaksi. Item keempat adalah item 557. Item 557 telah dibeli pada 3,13% dari total transaksi yang dilakukan di minimarket K1mart yaitu sebanyak 309 transaksi. Dan pada item 39, item ini telah dibeli pada 3,06% dari keseluruhan transaksi yaitu sebanyak 303 transaksi.

Berbeda dengan *frequent* item pada 1 *itemset* barang, pada 2 *itemset* barang peningkatan nilai *support* pada barang yang telah digeneralkan hanya sebesar 0,0006. Pada 2 *itemset* terdapat 23 item yang *frequent*. Berikut ini adalah 5 item dengan nilai *support* tertinggi.

Tabel 4.13 2 *Itemset* dengan *Support* Tertinggi pada Data Barang Umum Algoritma Apriori

No	Item 1	Keterangan Item 1	Item 2	Keterangan Item 2	Support
1	1299	Pucuk Harum Minuman Teh Melati Btl	256	Club Air Mineral Btl	0,0035
2	1057	Mr.Bread Roti Manis Isi Coklat Special Pck	256	Club Air Mineral Btl	0,0031
3	1646	Ultra Susu Uht Steril Slim Chocolate Tpk	256	Club Air Mineral Btl	0,0030
4	256	Club Air Mineral Btl	72	Aqua Air Mineral Prima Btl	0,0029
5	1255	Pocari Sweat Minuman Isotonik Btl	256	Club Air Mineral Btl	0,0028

Pada 2 *itemset* yang *frequent*, nilai *support* untuk item 1299 dan 256 adalah sebesar 0,0035. Artinya item 1299 dan item 256 telah dibeli bersama-sama dalam 0,35% dari semua transaksi yaitu sebanyak 35 kali. Item 1057 dan item 256 dibeli bersama-sama sebanyak 31 kali atau 0,31% dari total transaksi. Untuk item 1646 dan item 256, item ini telah dibeli secara bersama-sama dalam 0,30% dari transaksi yaitu sebanyak 30 kali. Nilai *support* untuk dua pasang item selanjutnya tidak jauh berbeda dengan nilai *support* pada item 1646 dan item 256. Pada item 256 dan item 72 nilai *support* nya adalah 0,0029. Artinya 0,29% dari total transaksi telah terdapat item 256 dan 72 di dalamnya. Jadi dari 9871 transaksi, terdapat 29 transaksi yang mengandung item 256 dan 72. Pasangan item terakhir pada 2 *itemset* dengan nilai *support* tertinggi adalah item 1255 dan item 256. Item ini dibeli dalam 0,28% dari transaksi atau sebanyak 28 kali.

Berdasarkan algoritma apriori, *itemset* maksimum yang *frequent* adalah dua item. Sedangkan pada 3 *itemset* sudah tidak ada yang memenuhi minimum *support* yang telah ditetapkan. Berdasarkan 298 *frequent item* di atas, terdapat 4133 *rule* yang dihasilkan. Berikut ini 10 *rule* dengan nilai *support* tertinggi.

Tabel 4.14 Rule pada Data Barang Umum Algoritma Apriori

No	LHS	RHS	Support	Confidence	Lift
1	1299 Pucuk Harum Minuman Teh Melati Btl	256 Club Air Mineral Btl	0,0035	0,1598	1,0524
2	1057 Mr.Bread Roti Manis Isi Coklat Special Pck	256 Club Air Mineral Btl	0,0031	0,1751	1,1533
3	1646 Ultra Susu Uht Steril Slim Chocolate Tpk	256 Club Air Mineral Btl	0,0030	0,1003	0,6607
4	1255 Pocari Sweat Minuman Isotonik Btl	256 Club Air Mineral Btl	0,0028	0,1029	0,6779
5	71 Aqua Air Mineral Btl	256 Club Air Mineral Btl	0,0027	0,0665	0,4379
6	424 Floridina Juice Pulp Orange Btl	256 Club Air Mineral Btl	0,0025	0,1256	0,8272
7	1057 Mr.Bread Roti Manis Isi Coklat Special Pck	1646 Ultra Susu Uht Steril Slim Chocolate Tpk	0,0025	0,1412	4,6629

Tabel 4.14 Rule pada Data Barang Umum Algoritma Apriori
(Lanjutan)

No	LHS	RHS	Support	Confidence	Lift
8	1646 Ultra Susu Uht Steril Slim	1057 Mr.Bread Roti Manis Isi Coklat Special Pck	0,0025	0,0836	4,6629
9	1646 Ultra Susu Uht Steril Slim	72 Aqua Air Mineral Prima Btl	0,0022	0,0735	1,1037
10	103 Bear Brand Susu Encer Steril K1g	256 Club Air Mineral Btl	0,0022	0,1618	1,0652

Pada 10 *rule* yang telah tercantum pada Tabel di atas, terdapat empat *rule* yang memiliki nilai *lift* kurang dari 1. *Rule* tersebut antara lain pada *rule* ke tiga, *rule* ke empat, *rule* ke lima, dan *rule* ke enam. Artinya hubungan antara LHS dan RHS pada *rule-rule* tersebut tidak signifikan.

Berdasarkan nilai *lift* pada Tabel, didapatkan bahwa *rule* yang paling signifikan adalah *rule* ke tujuh dan *rule* ke delapan. *Rule* ketujuh yaitu jika konsumen membeli item 1057 maka dia akan membeli item 1646 dengan nilai *lift* 4,6629. Pada *rule* ke tujuh ini nilai *support* yang didapatkan adalah sebesar 0,0025 dan nilai *confidence* adalah sebesar 0,1412. Artinya terdapat 0,25% atau sebesar 25 transaksi yang membeli item 1057 dan item 1646 secara bersamaan dan 14,12% dari transaksi yang membeli item 1057 juga membeli item 1646.

Rule ke delapan yaitu jika konsumen membeli item 1646 maka konsumen tersebut juga akan membeli item 1057. Nilai *support* yang didapatkan pada *rule* 8 ini sama dengan nilai *support* pada *rule* tujuh dan nilai *lift* nya pun sama. Perbedaannya hanya pada nilai *confidence* dan penempatan item pada LHS dan

RHS saja. Berdasarkan nilai *confidence*, kepercayaan konsumen yang membeli item 1646 juga akan membeli item 1057 hanya sebesar 8,36% saja. Jadi dari *rule* tujuh dan *rule* delapan, dengan nilai *support* dan *lift* yang sama dan item yang digunakan dalam membentuk *rule* juga sama maka *rule* yang lebih menarik adalah *rule* tujuh. *Rule* tersebut yaitu konsumen membeli 1057 maka konsumen akan membeli 1646 dengan nilai *confidence* yang lebih tinggi yaitu 14,12%.

Pada *rule* 1, item 1299 dan item 256 telah dibeli secara bersama-sama didalam 0,35% transaksi atau 35. Dimana dari pembelian item 1299, terdapat 15,98% transaksi yang juga dilakukan untuk membeli item 256.

Rule dua yaitu jika konsumen membeli 1057 maka konsumen tersebut juga akan membeli 2556. Pada *rule* ini item 1057 dan item 2556 telah dibeli secara bersama-sama dalam 0,31% transaksi yaitu sekitar 31 kali. Tingkat kepercayaan seorang yang membeli item 1057 juga akan membeli item 2556 adalah 17,51% dan nilai *lift* yang dihasilkan lebih dari satu jadi peningkatan pembelian pada item 1057 akan meningkatkan pembelian pada item 2556 juga.

Pada *rule* ke sembilan yang terbentuk, jika konsumen membeli item 1646 maka konsumen tersebut juga akan membeli item 256. Item 1646 dan item 256 telah dibeli secara bersama-sama dalam 0,225 dari total transaksi dan tingkat kepercayaan jika konsumen membeli item 1646 maka konsumen tersebut juga akan membeli item 256 adalah sebesar 7,35%.

Rule sepuluh yaitu jika konsumen membeli item 103 maka akan membeli item 256. Terdapat 0,22% konsumen yang membeli item 103 dan 256 secara bersamaan. Dimana dari semua pembelian item 103, terdapat item 256 yang juga dibeli didalamnya.

4.5 Pola Perilaku Konsumen dengan Algoritma Squeezer

Pada pembahasan sebelumnya telah dianalisis mengenai pola pembelian konsumen pada minimarket K1mart menggunakan metode *association rule* dengan algoritma FP-

Growth, dan *Apriori*. Pada bab ini akan dilakukan analisis mengenai pola pembelian konsumen menggunakan salah satu metode dalam *clustering* yaitu algoritma *squeezer*. Data yang digunakan dalam analisis ini adalah data transaksi penjualan dari minimarket *K1mart*. Data transaksi penjualan yang digunakan terdiri dari 9871 transaksi.

Sebelum dilakukan *clustering* dengan algoritma *squeezer*, dilakukan pengurangan jumlah data yang akan dikelompokkan dengan menggunakan minimum *support* 0,0015. Setelah dilakukan pengurangan jumlah item dengan minimum *support*, jumlah data barang detail yang dikelompokkan berkurang menjadi 381 item saja dan data barang umum menjadi sejumlah 275 item.

4.5.1 Data Barang Detail

Pada data barang dengan nama detail dari 381 item terbentuk 379 cluster. Jadi hanya terdapat dua *cluster* yang masing-masing berisi dua macam item, yang lainnya di dalam satu *cluster* yang terbentuk hanya terdapat satu item saja. Berikut ini adalah daftar *cluster* yang memiliki anggota lebih dari satu item.

Tabel 4.15 Hasil *Cluster* Data Barang Detail Algoritma *Squeezer*

Cluster	Anggota 1	Anggota 2	Similarity
33	Item 168 0060410 Indomi Goreng Spc 80	Item 84 0013550 Indomi Grg Pedas 80g	0,000124
59	Item 753 1153830 Soba Mi Stik S.Bld24	Item 752 1153820 Soba Mi Stik A.Bkr24	0,000229

Pada algoritma *squeezer* hanya didapatkan dua pasang barang yang sering dibeli secara bersama-sama oleh konsumen untuk data barang detail. Item yang sering dibeli secara bersama-sama pada data barang detail adalah item 168 dengan item 82 dan item 753 dengan item 752. Berdasarkan Tabel 4.17 dapat dilihat

bahwa nilai *similarity* pada *cluster* 59 lebih besar dari *cluster* 33. Jadi dapat dikatakan bahwa item 753 dan item 752 lebih sering dibeli secara bersama-sama jika dibandingkan dengan item 168 dan item 84.

4.5.2 Data Barang Umum

Pada *clustering* data barang umum, terdapat 275 item yang dikelompokkan. Berikut ini adalah hasil *clustering* untuk barang umum pada *cluster* dengan jumlah anggota lebih dari satu.

Tabel 4.16 Hasil *Cluster* pada Data Barang Umum Algoritma *Squeezer*

Cluster	Anggota 1	Anggota 2	Similarity
34	637 Indomie Mie Goreng Plus Special Pck	636 Indomie Mie Goreng Pedas Pck	0,000124
58	1480 Soba Mie Stick Ala Jepang Premium Sambal Balado Pck	1479 Soba Mie Stick Ala Jepang Premium Ayam Bakar Pck	0,000229

Berdasarkan Tabel 4.18, dapat dilihat bahwa dari 173 *cluster* yang terbentuk hanya ada dua *cluster* yang memiliki lebih dari satu anggota. *Cluster* tersebut antara lain *cluster* 34 dan *cluster* 58. *Cluster* 34 memiliki dua anggota yaitu item 637 dan item 636. Nilai *similarity* dari *cluster* 34 adalah sebesar 0,000124. Sama dengan *cluster* 34, *cluster* 58 juga memiliki dua anggota saja di dalamnya. Anggota dari *cluster* 58 adalah item 1480 dan item 1479. Nilai *similarity* yang dimiliki *cluster* 58 lebih tinggi dari *cluster* 34 yaitu sebesar 0,000229. Jadi dapat dikatakan bahwa item pada *cluster* 58 lebih sering dibeli secara bersama-sama jika dibandingkan dengan *cluster* 34.

4.6 Perbandingan Algoritma *FP-Growth*, *Apriori* dan *Squeezer*

Pada analisis perilaku konsumen di minimarket K1mart ITS, terdapat tiga algoritma yang digunakan yaitu *FP-Growth*, *Apriori*, dan *Squeezer*. Perbandingan dari ketiga algoritma ini dapat dilihat di dalam Tabel 4.19 berikut ini.

Tabel 4.17 Perbandingan Algoritma *FP-Growth*, *Apriori*, dan *Squeezer*

Algoritma	Data Barang Detail		Data Barang Umum	
	Waktu	Hasil	Waktu	Hasil
<i>FP-Growth</i>	39 seconds	9 rule	34 seconds	17 rule
<i>Apriori</i>	0,01 seconds	4004 rule	0,00 seconds	4133 rule
<i>Squeezer</i>	15,175 3 hours	2 pola pembelian	8,93059 hours	2 pola pembelian

Berdasarkan Tabel 4.17 dapat diketahui bahwa jumlah pola pembelian yang dihasilkan untuk setiap algoritma berbeda. Dengan menggunakan nilai minimum *support* yang sama yaitu 0,0015, algoritma yang menghasilkan paling sedikit pola pembelian adalah algoritma *squeezer* dan yang menghasilkan jumlah pola pembelian paling banyak adalah algoritma *Apriori*. Meskipun algoritma *squeezer* menghasilkan paling sedikit pola pembelian, namun waktu proses yang dihasilkan lebih lama dibandingkan dengan algoritma yang lain. Sedangkan algoritma yang memiliki waktu proses yang paling sedikit adalah algoritma *Apriori*.

Perbedaan lain yang dihasilkan dari ketiga algoritma ini adalah pada algoritma *apriori* dan *FP-Growth* analisis yang dihasilkan adalah jika konsumen membeli item A maka konsumen juga akan membeli item B. Jadi dapat diketahui item apakah yang dibeli terlebih dahulu. Namun pada algoritma *squeezer* analisis yang dihasilkan hanyalah item yang sering dibeli secara bersama-sama saja. Jadi tidak bisa diketahui item manakah yang dibeli terlebih dahulu.

Algoritma yang baik adalah yang mampu menghasilkan *rule* dengan nilai *support*, *confidence*, dan *lift* yang tinggi. Berdasarkan hasil analisis yang telah dibahas, algoritma *FP-Growth* mampu menghasilkan *rule* dengan nilai *support*, *confidence*, dan *lift* yang lebih tinggi. Sedangkan pada algoritma *squeezer* tidak memiliki ketiga nilai *interesting* ini sehingga tidak dapat dibandingkan dengan algoritma *apriori*, dan *FP-Growth*.

4.7 Letak Barang yang Sering Dibeli Konsumen di Minimarket *K1mart*

Pada analisis perilaku konsumen di minimarket *K1mart* menggunakan algoritma *FP-Growth* pada data barang detail, jika konsumen membeli Soba Mi Stik A.Bkr24 maka konsumen juga akan membeli Soba Mi Stik S.Bld24. Pada algoritma *apriori*, pola perilaku konsumen yang didapatkan adalah jika konsumen membeli Mr. Bread Roti Cokl160 maka konsumen juga akan membeli Ultra Uht Choco 250m. Sedangkan untuk algoritma *squeezer*, Soba Mi Stik S.Bld24 dan Soba Mi Stik A.Bkr24 sering dibeli oleh konsumen *K1mart* secara bersama-sama. Berikut ini adalah gambar item-item tersebut pada minimarket *K1mart*.

Analisis yang dihasilkan pada data barang umum tidak jauh berbeda dengan analisis yang dihasilkan pada data barang detail. Pada algoritma *FP-Growth*, pola perilaku konsumen yang dihasilkan adalah jika konsumen membeli Soba Mie Stick Ala Jepang Premium Ayam Bakar Pck maka konsumen juga akan membeli Soba Mie Stick Ala Jepang Premium Sambal Balado Pck. Hasil dari algoritma *apriori* adalah pembelian jika konsumen membeli Mr.Bread Roti Manis Isi Coklat Special Pck maka konsumen juga akan membeli Club Air Mineral Btl. Sedangkan untuk algoritma *squeezer* pola perilaku konsumen yang dihasilkan adalah Soba Mie Stick Ala Jepang Premium Sambal Balado Pck dan Soba Mie Stick Ala Jepang Premium Ayam Bakar Pck sering dibeli secara bersama-sama.



Gambar 4.10 Letak Roti pada Minimarket *K1mart*



Gambar 4.11 Letak Soba Mi pada Minimarket *K1mart*



Gambar 4.12 Letak Susu Ultra pada Minimarket *K1mart*



Gambar 4.13 Letak Club Air Mineral pada Minimarket *K1mart*

Pada algoritma *apriori*, pola perilaku konsumen yang didapatkan adalah jika konsumen membeli Mr. Bread Roti Cokl60 maka konsumen juga akan membeli Ultra Uht Choco 250m. Pada Gambar 4.10 dan Gambar 4.12 terlihat bahwa Mr. Bread Roti Cokl60 dan Ultra Uht Choco 250m terletak pada rak yang berbeda. Seharusnya Mr. Bread Roti Cokl60 dan Ultra Uht Choco 250m diletakkan pada rak yang sama.

Pola perilaku konsumen lainnya yang didapatkan yaitu pada algoritma *apriori*, jika konsumen membeli Soba Mi Stik A.Bkr24 maka konsumen juga akan membeli Soba Mi Stik S.Bld24. Berdasarkan Gambar 4.11, ternyata Soba Mi Stik A.Bkr24 dan Soba Mi Stik S.Bld24 sudah diletakkan pada satu rak yang sama dan letaknya berdekatan.

Pola perilaku konsumen selanjutnya yang didapatkan yaitu pembelian jika konsumen membeli Mr.Bread Roti Manis Isi Coklat Special Pck maka konsumen juga akan membeli Club Air Mineral Btl. Berdasarkan Gambar 4.10 dan 4.13 terlihat bahwa kedua item ini diletakkan pada rak yang berbeda.

4.8 Pola Perilaku Konsumen yang Dihasilkan pada Item dengan Support Tertinggi

Item yang memiliki nilai *support* tertinggi pada data barang detail adalah item 589 dengan nilai *suppot* sebesar 0,0965. Sedangkan untuk data barang umum, item yang memiliki nilai *support* tertinggi adalah item 256 dengan nilai *support* sebesar 0,1518.

Pada *rule-rule* yang dihasilkan pada algoritma *FP-Growth*, terdapat tiga *rule* yang mengandung item 589. Sedangkan pada data umum, dengan menggunakan algoritma *FP-Growth* terdapat lima *rule* yang mengandung item 256. Pada data barang detail, terdapat empat *rule* dari sepuluh *rule* dengan nilai *support* tertinggi yang dihasilkan pada algoritma *apriori* yang mengandung item 589. Pada data barang umum, terdapat tujuh *rule* yang mengandung item 256 pada *rule* yang dihasilkan dengan algoritma *apriori*. Namun pada algoritma *squeezer* baik pada analisis data detail tidak ada pola pembelian konsumen yang

mengandung item 589. Pada analisis data barang umum dengan menggunakan algoritma *squeezer* juga tidak ada pola pembelian yang dihasilkan yang mengandung item 256.

(halaman ini sengaja dikosongkan)

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Setelah dilakukan analisis dan juga pembahasan pada Bab IV, maka berikut ini merupakan kesimpulan didapatkan pada analisis perilaku konsumen di minimarket *K1mart*.

1. Pada algoritma *FP-Growth* terdapat sembilan pola pembelian konsumen yang didapatkan pada data barang detail. Pola pembelian konsumen dengan nilai *confidence* tertinggi yaitu jika konsumen *K1mart* membeli Soba Mi Stik A.Bkr24 maka konsumen *K1mart* juga akan membeli Soba Mi Stik S.Bld24. Sedangkan pada data barang umum didapatkan 17 pola pembelian konsumen. Pola pembelian dengan nilai *confidence* tertinggi yaitu jika konsumen *K1mart* membeli Soba Mie Stick Ala Jepang Premium Ayam Bakar Pck maka konsumen *K1mart* juga akan membeli Soba Mie Stick Ala Jepang Premium Sambal Balado Pck.
2. Pada algoritma apriori didapatkan 4007 *rule* yang dihasilkan pada data barang detail dan 4135 *rule* yang dihasilkan pada data barang umum. Pola pembelian dengan nilai *confidence* tertinggi pada data barang detail adalah jika konsumen *K1mart* membeli Mr. Bread Roti Cokl60 maka konsumen juga akan membeli Ultra Uht Choco 250m. Sedangkan pada data barang umum didapatkan pola pembelian jika konsumen *K1mart* membeli Mr.Bread Roti Manis Isi Coklat Special Pck maka konsumen juga akan membeli Club Air Mineral Btl.
3. Pada algoritma *squeezer* didapatkan dua pola pembelian saja baik pada data barang detail maupun data barang umum. Pola pembelian pada data barang detail adalah Soba Mi Stik S.Bld24 dan Soba Mi Stik A.Bkr24 sering dibeli oleh konsumen *K1mart* secara bersama-sama. Sedangkan pada data barang umum didapatkan pola pembelian konsumen

yaitu item Soba Mie Stick Ala Jepang Premium Sambal Balado Pck dan Soba Mie Stick Ala Jepang Premium Ayam Bakar Pck sering dibeli secara bersama-sama.

4. Algoritma yang memiliki waktu proses paling cepat adalah algoritma *Apriori*.
5. Algoritma terbaik yang didapatkan adalah *FP-Growth*, sebab mampu menghasilkan *rule* dengan nilai *support*, *confidence*, dan *lift* yang lebih besar jika dibandingkan dengan algoritma *Apriori*.
6. Algoritma *FP-Growth* dan *Apriori* mampu menghasilkan barang apa yang dibeli konsumen terlebih dahulu dan barang apakah yang akan dibeli selanjutnya. Algoritma *FP-Growth* dan *apriori* juga mampu menghasilkan lebih dari satu pola perilaku konsumen untuk satu jenis barang. Sedangkan algoritma *squeezer* tidak bisa keduanya.

5.2 Saran

Berdasarkan analisis yang telah dilakukan, terdapat beberapa saran yang diberikan untuk mengembangkan minimarket *K1mart*.

1. Item-item yang sering dibeli bersama-sama oleh konsumen diletakkan pada tempat yang berdekatan agar mampu meningkatkan penjualan.
 2. Membuat paket promosi untuk item-item yang sering dibeli secara bersama-sama oleh konsumen.
- Beberapa saran yang diberikan untuk perbaikan maupun pengembangan penelitian ini lebih lanjut antara lain.
1. Pada pencarian item yang sering dibeli konsumen secara bersama-sama, sebaiknya menggunakan metode *association rule*. Sebab jika menggunakan metode *clustering* maka satu barang hanya bisa dimasukkan ke dalam satu *cluster* saja.
 2. Untuk meningkatkan nilai kebaikan model yang dihasilkan, disarankan menggunakan data pembelian pada *retail* yang digunakan sebagai tempat belanja bulanan konsumen.

3. Membuat *group forecasting* pada item yang sering dibeli bersama-sama oleh konsumen untuk menentukan jumlah persediaan barang.

(halaman ini sengaja dikosongkan)

DAFTAR PUSTAKA

- Barskey, K. (1992). *Beyond Customer Satisfaction To Customer Loyalty*. New York: AMA Management Briefing.
- Berry, M. J., & Linoff, G. S. (2004). *Data Mining Technique For Marketing, Sales, And Customer Relationship Management*. United States Of America: Wiley.
- Borgelt, C. (2005) *An Implementation Of The FP-Growth Algorithm* [Online]. Diambil dari: <http://www.borgelt.net/papers/fpgrowth.pdf>. [Diakses pada : 2-Sep-2016].
- Brosekhan, A. A., & Velayutham, C. M. Customer Buying Behaviour-A Literature Review. *IOSR Journal Business And Management (IOSR-JBM)*, 8-16.
- Failasufa, F. (2014) *Penerapan Data Mining Analisis Pola Pembelian Konsumen Dengan Algoritma Apriori Pada Data Transaksi Penjualan* [Online]. Diambil dari: <http://digilib.uin-suka.ac.id/13276/1/BAB%20I.%20V.%20DAFTAR%20PUSTAKA.pdf> [Diakses pada 4-Sep-2016].
- Fatihatul, F., Setiawan, A., & Rosadi, R. (2011) *Asosiasi Data Mining Menggunakan Algoritma FP-Growth Untuk Market Basket Analysis* [Online]. Diambil dari : https://www.academia.edu/4758451/ASOSIASI_DATA_MINING_MENGGUNAKAN_ALGORITMA_FP-GROWTH_UNTUK_MARKET_BASKET_ANALYSIS [Diakses pada: 4-Sep-2016].
- Han, J., & Kember, M. (2006). *Data Mining Concepts And Techniques*. United States Of America: Morgan Kaufmann.
- Han, J., Pei, J., Yin, Y., & Mau, R. (2004). Mining Frequent Pattern Without Candidate Generation : A Frequent Pattern Tree. *Data Mining And Knowledge Discovery*, 54-87.

- Kumar, N. (2003). Kill A Brand, Keep A Customer. *Harvard Bussiness Review*, 86-95.
- Larose, D. T., & Larose, C. D. (2015). *Data Mining And Predictive Analytics*. United States Of America: Wiley.
- Limanto, S. (2005). Clustering Untuk Analissi Market Basket. *Statistika*. Surabaya : Institut Teknologi Sepuluh Nopember.
- Lussy, S. (2016) *Jumlah Perusahaan Ritel Di Indonesia Menghawatirkan* [Online]. Diambil dari: [Http://Sentananews.Com/News/Ekonomi/Jumlah-Perusahaan-Ritel-Di-Indonesia-Mengkhawatirk-12582](http://Sentananews.Com/News/Ekonomi/Jumlah-Perusahaan-Ritel-Di-Indonesia-Mengkhawatirk-12582) [Diakses pada : 1-Sep-2016].
- Porkodi, R., & P., P. (2015). A Comparative Analysis Of Assosiation Rule Mining Algorithms In Data Mining : A Study. *American Journal Of Computer Science And Engineering Survey*, 98-119.
- Raorane, A., & Kulkarni, R. V. (2011). Data Mining Techniques : A Source For Consumer Behavior Analysis. *International Journal Of Database Management Systems (IJDMS)*, Vol 3 No.3, 45-56.
- Samuel, D. (2008) *Penerapan Struktur FP-Tree dan Algoritma FP-Growth dalam Optimasi Penentuan Frequent Itemset* [Online]. Diambil dari: <http://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2007-2008/Makalah/MakalahIF2153-0708-006.pdf> [Diakses pada : 16-Sep-2016].
- Timm, P. (2005) *50 Ideas To Keep Your Customer* [Online]. Diambil dari : <https://books.google.co.id/books?id=CUtOuUl5XOYC&pg=PA1&lpg=PA1&dq=tim+50+Ideas+To+Keep+Your+Customer&source=bl&ots=c8MEFKPPpA&sig=dxPHVnkU6FiW93hS7UhYXvtRtpQ&chl=en&sa=X&ved=0ahUKEwi9tMDZ8L3RAhVJLI8KHUx3AOgQ6AEIKzAd#v=onepage&q=tim+50+Ideas+To+Keep>

20Your%20Customer&f=false [Diakses pada: 28-Sep-2016].

Venkatachari, K. (2016). Market Basket Analysis Using FP Growth Apriori Algorithm : A Case Study Of Mumbai Retail Store. *BVIMR's Journal Of Management Research*, 56-63.

Victor, H. A., Abimbola, O. O., Mercy, O. F., Esther, O. O., & Eloho, I. P. (2014). Customer Behavior Analytics And Data Mining. *American Journal Of Computation, Communication And Control*, 66-74.

Yusuf, M. H., & Kuncoro, A. J. (2014) *Analisa Peningkatan Penjualan Produk Menggunakan Metode Market Basket Analysis Pada Perbaikan Layout Dalam Association Rules Di Minimarket Alfamart Jalan Damai, Sleman, Yogyakarta* [Online]. Diambil dari : <https://www.scribd.com/doc/232573823/market-basket-analysis-swalayan> [Diakses pada: 7-Sep-2016].

(halaman ini sengaja dikosongkan)

LAMPIRAN

Lampiran 1. Contoh Struk Transaksi

@ K1 MART
KOPERASI PEGAWAI NEGERI ITS
KAMPUS ITS JL ARIEF RAHMAN HAKIM
KEL.KEPUTIH
KEC.SUKOILLO
SURABAYA
Telp.

===== INISIALISASI =====
Station : 02 Tanggal : 01-03-2016
Shift : 1 Jam : 07:54:54
----- INISIALISASI -----

ID Kasir : IZA
Nama Kasir : IZA
Cash Awal : Rp. 200.000

===== INISIALISASI =====

: 01-03-16/07:57:05 O385/00001/1/IZA

1142400 YOU C1000 ORG WTR500
1 7.000 0 7.000
1250800 FF LOW FAT VAN 225
1 5.000 0 5.000

Item/Qty (2/ 2)
HARGA JUAL.....: 12.000
PEMBAYARAN TUNAI : 13.000
KEMBALI.....: 1.000

TERIMA KASIH
BARANG YANG SUDAH DI BELI
TIDAK DAPAT DI KEMBALIKAN LAGI

: 01-03-16/08:08:48 O385/00002/1/IZA

0357300 234 ROKOK KRT 12/PCS **
1 14.100 0 14.100
0357180 GG SURYA FILTER 12'S **
1 13.500 0 13.500

Item/Qty (2/ 2)
HARGA JUAL.....: 27.600
PEMBAYARAN TUNAI : 50.000
KEMBALI.....: 22.400

TERIMA KASIH
BARANG YANG SUDAH DI BELI
TIDAK DAPAT DI KEMBALIKAN LAGI

Lampiran 2. Database Pembelian Barang Detail

Kode	Quantity	Transaction
725	1	1
962	1	1
317	1	2
310	1	2
845	1	3
589	1	3
589	1	4
651	2	5
168	2	6
137	1	6
83	1	6
541	1	6
879	1	6
1230	1	7
⋮	⋮	⋮
1282	1	9864
343	1	9865
1175	1	9865
400	1	9865
179	1	9866
40	1	9866
959	1	9867
993	1	9868
57	1	9869
671	1	9870
449	1	9870
59	1	9870

Lampiran 3. Database Pembelian Barang Umum

Kode	Quantity	Transaction
949	1	1
1254	1	1
407	1	2
400	1	2
1106	1	3
762	1	3
762	1	4
842	2	5
212	2	6
174	1	6
107	1	6
699	1	6
1145	1	6
1641	1	7
:	:	:
434	1	9865
1562	1	9865
516	1	9865
229	1	9866
53	1	9866
1251	1	9867
1298	1	9868
74	1	9869
869	1	9870
571	1	9870
77	1	9870
409	1	9871

Lampiran 4. *Syntax* Membaca Notepad

```
ListNamaFile = read.table("D:/NamaFileQoqom.txt",header=FALSE)
m.pembelian = data.frame()
notran = 0
j = 1

for (nofile in 1:nrow(ListNamaFile))
{
 >NamaFile=paste0("D:/TUGAS
AKHIR/STRUK/0",ListNamaFile[nofile,1],".txt")
RawFile = read.delim>NamaFile, header=FALSE)
n = nrow(RawFile)
# Cleaning un-need row
ambil=c()
for (i in 1:n)
{
  if (substr(RawFile[i,1],3,3) == "#" || substr(RawFile[i,1],2,2) == ":")
  {ambil = c(ambil,i)}
}
# Membuat tabel pembelian barang
for (i in 1:length(ambil))
{
  if (substr(RawFile[ambil[i],1],2,2) != ":")
  {
    m.pembelian[j,1] = as.character(RawFile[ambil[i]-1,1])
    m.pembelian[j,2] = as.numeric(substr(RawFile[(ambil[i]),1],4,12))
    m.pembelian[j,3] = notran
    j=j+1
  }else
  {
    notran = notran+1
  }
}
print(paste0>NamaFile," telah selesai diproses"))
}
colnames(m.pembelian) <- c("Name","Quantity","Transaction")
write.csv(m.pembelian,"D:/DataBulanMaretpembelian.csv",
row.names = FALSE)
```


Lampiran 5. Syntax Menulis Barang Berdasarkan Transaksi

```
data=read.csv("D://DataBulanMaretpembelianDeleteKantong.csv",
sep=";", header=T)
library(plyr)
library(dplyr)
#menggabungkan transaksi yang sama
datapembelian1=ddply(data, c("Name"),
function(df1)paste(df1$Transaction,collapse = ","))
#menyimpan data dengan struktur apriori
write.table(datapembelian1,"D://DaftarBarangMaretpembelianDelete
Kantong.csv",sep=";", col.names=FALSE ,row.names=TRUE)
```

Lampiran 6. Syntax Menulis Data dalam Bentuk *Market Basket*

```
m.pembelian=read.csv("D://4.DataMaretDetailKode.csv",sep=";",he
ader=TRUE)
# Membuat tabel pembelian
ntran = table(m.pembelian[,3])
nname = rownames(table(m.pembelian[,1]))
notran=nrow(ntran)
m.data = data.frame(matrix(0, nrow = notran, ncol =
(length(nname)+1)))
colnames(m.data) =
c("Transaction",rownames(table(m.pembelian[,1])))
temp = 1
for (i in 1:notran)
{
for (j in 1:ntran[i])
{
m.data[i,1] = i
m.data[i,(which(nname == m.pembelian[temp,1])+1)] = 1
temp=temp+1
}
}
write.csv(m.data,"D://xxxBiner4.DataMaretDetailKode.csv",
row.names = FALSE)
```

Lampiran 7. *Syntax Algoritma Apriori*

```
library(arules)
library(arulesViz)
data=read.csv("D://1.1.KodeMaretDetailpembelian.csv", sep=";",
header=T)
transd=read.transactions(file="D://1.1.KodeMaretDetailpembelian.c
sv",format="single",sep=";",cols=c("Transaction","Kode"))

#frequentitem
#ini untuk melihat item yang frequent 1 itemset
items=apriori(transd,parameter=list(target="frequent",supp=0.0015,
minlen=1,maxlen=1))
a1=as(sort(items),"data.frame")
write.csv(a1,"D:/DataMaretDetail
minsup0.0015item1.csv",row.names=FALSE)
inspect(sort(items, by='support', decreasing = T)[1:5])

#ini untuk 2 itemset
items2=apriori(transd,parameter=list(target="frequent",supp=0.001
5,minlen=2,maxlen=2))
a2=as(sort(items2),"data.frame")
write.csv(a2,"D:/DataMaretDetail
minsup0.0015item2.csv",row.names=FALSE)
inspect(sort(items2, by='support', decreasing = T)[1:5])

#ini untuk semua itemset
items0=apriori(transd,parameter=list(target="frequent",supp=0.001
5,minlen=1))
a0=as(sort(items0),"data.frame")
write.csv(a0,"D:/DataMaretDetail
minsup0.0015allitem.csv",row.names=FALSE)
inspect(sort(items0, by='support', decreasing = T)[1:10])

#membuat rule
rulesapriori=apriori(transd, parameter = list(supp
=0.0015,confidence=0.05,minlen=1, originalSupport=FALSE))
a=as(rulesapriori,"data.frame")
write.csv(a,"D:/DataDetailminsup0,0015minlen1.csv",row.names=F
ALSE)
```


Lampiran 8. *Syntax Algoritma Squeezer*

```

data=read.csv("D://3.1.MaretUmumSqueezerminsup=0.0015.csv",se
p=";",header=FALSE)
mdataok=as.matrix(data)
#nilai harus diinput
beta=1
alfa=1
minsim=0.00001
baris=nrow(mdataok)
kolom=ncol(mdataok)
ntotal=baris-1
#inisiasi untuk cluster1
cluster=data.frame()
cluster=mdataok[,1]
cluster=as.matrix(cluster)
cluster=cbind(cluster)
  ncluster=1
  daftarcluster=c("barang","similarity","cluster")
  for (k in 2 :kolom )
  {
    k=5
    sim=data.frame()
    baranga=c()
    barangb=c()
    barangab=data.frame()
    barangb=mdataok[,k]    #ngitungbarang b
    nb=sum(barangb[2:baris])
    for (d in 1 :ncluster)
    {
      baranga=cluster[,d]    #ngitung barang a
      na=sum(baranga[2:baris])
      for (ab in 1:baris)
      {
        if (baranga[ab]==barangb[ab])
          barangab[ab,d]=barangb[ab] else
          barangab[ab,d]=0
      }
    }
  }

```

Lampiran 8. *Syntax* Algoritma Squeezer (Lanjutan)

```

nab=sum(barangab[2:baris,d])
#ngitung similarity
x=(nab^2)/(na*nb)
y=nab/ntotal
sim[d,1]=(x^alfa)*(y^beta)
sim[d,2]=d

}
simurut=sim[order(sim[,1]),] #iniurutan dari kecil ke besar
similarity=simurut[d,1]
dmaks=simurut[d,2]
if (similarity<minsims)
{
cluster=cbind(cluster,barangb)
ncluster=ncluster+1

barangcluster=c(barangb[1],simurut[d,1],ncluster)
#barang,similarity,cluster
daftarcluster=rbind(daftarcluster,barangcluster)

}else
{

cluster[,dmaks]=barangab[,dmaks]
barangcluster=c(barangb[1],simurut[d,1],simurut[d,2])
#barang,similarity,cluster
daftarcluster=rbind(daftarcluster,barangcluster)

}
}
write.table(daftarcluster,"D:/3.1.MaretUmumSqueezerminsu
p=0.0015minsims=0.00001.csv",sep="," ,
col.names=FALSE,row.names = FALSE)

```


Lampiran 9. *Output* Algoritma FP-Growth Data Barang Detail
1 Itemset

No	Item	Support	No	Item	Support
1	589	0,097	28	1228	0,01
2	80	0,067	29	774	0,009
3	590	0,058	30	567	0,009
4	79	0,041	31	1116	0,009
5	1045	0,031	32	893	0,009
6	123	0,03	33	168	0,009
7	571	0,027	34	1229	0,008
8	310	0,026	35	1001	0,008
9	937	0,022	36	725	0,008
10	1048	0,02	37	704	0,008
11	452	0,018	38	747	0,007
12	606	0,018	39	130	0,007
13	319	0,016	40	916	0,007
14	1282	0,015	41	640	0,007
15	962	0,014	42	651	0,007
16	1262	0,014	43	971	0,007
17	357	0,014	44	768	0,007
18	884	0,012	45	1285	0,007
19	420	0,012	46	856	0,006
20	761	0,011	47	605	0,006
21	963	0,011	48	40	0,006
22	641	0,011	49	558	0,006
23	64	0,011	50	726	0,006
24	124	0,011		⋮	
25	883	0,01	276	1222	0,002
26	400	0,01	277	1199	0,002
27	673	0,01	278	1150	0,002

Lampiran 10. *Output Algoritma FP-Growth Data Barang Detail 2 Itemset*

No	Item 1	Item2	Support
1	589	590	0,003
2	589	937	0,003
3	123	452	0,003
4	589	80	0,002
5	589	123	0,002
6	589	571	0,002
7	589	452	0,002
8	589	64	0,002
9	80	79	0,002
10	80	123	0,002
11	80	420	0,002
12	590	79	0,002
13	79	123	0,002
14	168	84	0,002
15	753	752	0,002

Lampiran 11. *Output Rule FP-Growth Barang Detail*

LHS	RHS	Support	Confidence	Lift
937	589	0,00284	0,12785	1,32429
420	80	0,00152	0,12931	1,93985
452	589	0,00243	0,13559	1,40445
452	123	0,00253	0,14124	4,66291
64	589	0,00162	0,14953	1,54883
168	84	0,00162	0,18182	47,22967
753	752	0,00162	0,29091	87,01708
84	168	0,00162	0,42105	47,22967
752	753	0,00162	0,48485	87,01708

Lampiran 12. *Output* Algoritma FP-Growth Data Barang Detail
1 Itemset

No	Item	Support	No	Item	Support
1	256	0,152	28	224	0,01
2	72	0,067	29	74	0,01
3	71	0,041	30	1648	0,01
4	557	0,031	31	1144	0,009
5	39	0,031	32	1092	0,009
6	1646	0,03	33	1012	0,009
7	1255	0,028	34	637	0,009
8	1118	0,027	35	815	0,008
9	1299	0,022	36	670	0,008
10	1376	0,02	37	1755	0,008
11	424	0,02	38	603	0,008
12	1057	0,018	39	448	0,007
13	1056	0,015	40	444	0,007
14	1597	0,015	41	1028	0,007
15	393	0,014	42	120	0,007
16	1298	0,014	43	1146	0,007
17	103	0,014	44	60	0,007
18	1649	0,013	45	1753	0,006
19	466	0,012	46	1147	0,006
20	838	0,012	47	1485	0,006
21	1651	0,012	48	1488	0,006
22	1511	0,012	49	573	0,006
23	110	0,011	50	1754	0,006
24	459	0,011		:	
25	445	0,011	273	1115	0,002
26	112	0,011	274	1066	0,002
27	462	0,01	275	1009	0,002

Lampiran 13. *Output* Algoritma *FP-Growth* Data Barang Detail 2
Itemset

No	Item1	Item2	Support
1	256	1299	0,004
2	256	72	0,003
3	256	71	0,003
4	256	1646	0,003
5	256	1255	0,003
6	256	424	0,003
7	256	1057	0,003
8	1646	1057	0,003
9	256	39	0,002
10	256	1118	0,002
11	256	1056	0,002
12	256	1298	0,002
13	256	103	0,002
14	256	1649	0,002
15	256	112	0,002
16	256	224	0,002
17	72	71	0,002
18	72	1646	0,002
19	72	1255	0,002
20	72	1651	0,002
21	71	1646	0,002
22	637	636	0,002
23	1480	1479	0,002

Lampiran 14. *Output Rule Data Barang Detail Algoritma FP-Growth*

No	LHS	RHS	Support	Confidence	Lift
1	1479	1480	0,00162	0,48485	87,01708
2	636	637	0,00162	0,42105	47,22967
3	1480	1479	0,00162	0,29091	87,01708
4	224	256	0,00203	0,20833	1,37189
5	637	636	0,00162	0,18182	47,22967
6	1057	256	0,00314	0,17514	1,15332
7	103	256	0,00223	0,16176	1,06523
8	1299	256	0,00355	0,15982	1,05241
9	112	256	0,00172	0,15888	1,04622
10	1057	1646	0,00253	0,14124	4,66291
11	1651	72	0,00152	0,12931	1,93985
12	424	256	0,00253	0,12563	0,82727
13	1649	256	0,00152	0,12097	0,79658
14	1298	256	0,00162	0,11679	0,76906
15	1056	256	0,00172	0,11333	0,74631
16	1255	256	0,00284	0,10294	0,67787
17	1646	256	0,00304	0,10033	0,66071

Lampiran 15. *Output Top 23 Frequent Itemset Barang Detail*
Algoritma Apriori untuk 1 Item

No	Items	Support
----	-------	---------

[1]	{589}	0,096545436
-----	-------	-------------

[2]	{80}	0,066659913
-----	------	-------------

[3]	{590}	0,058251444
-----	-------	-------------

[4]	{79}	0,041130585
-----	------	-------------

[5]	{1045}	0,030594671
-----	--------	-------------

[6]	{123}	0,030290751
-----	-------	-------------

[7]	{571}	0,026846317
-----	-------	-------------

[8]	{310}	0,026339783
-----	-------	-------------

[9]	{937}	0,022186202
-----	-------	-------------

[10]	{1048}	0,020160065
------	--------	-------------

[11]	{452}	0,017931314
------	-------	-------------

[12]	{606}	0,017627393
------	-------	-------------

[13]	{319}	0,015803870
------	-------	-------------

[14]	{1282}	0,015196029
------	--------	-------------

[15]	{962}	0,014182960
------	-------	-------------

[16]	{1262}	0,013879040
------	--------	-------------

[17]	{357}	0,013777733
------	-------	-------------

[18]	{884}	0,011852902
------	-------	-------------

[19]	{420}	0,011751596
------	-------	-------------

[20]	{761}	0,011447675
------	-------	-------------

[21]	{963}	0,011143754
------	-------	-------------

[22]	{641}	0,010941141
------	-------	-------------

[23]	{64}	0,010839834
------	------	-------------

Lampiran 16. *Output Frequent Itemset Data Barang Detail*
Algoritma Apriori untuk 2 Item

No	Items	Support
[1]	{589,590}	0,002937899
[2]	{589,937}	0,002836592
[3]	{123,452}	0,002532671
[4]	{452,589}	0,002431365
[5]	{123,80}	0,002228751
[6]	{123,589}	0,002228751
[7]	{590,79}	0,002026137
[8]	{589,80}	0,001924830
[9]	{752,753}	0,001620910
[10]	{168,84}	0,001620910
[11]	{589,64}	0,001620910
[12]	{571,589}	0,001620910
[13]	{123,79}	0,001620910
[14]	{420,80}	0,001519603
[15]	{79,80}	0,001519603

Lampiran 17. *Output Top 18 Rule Algoritma Apriori Data Barang Detail*

No	LHS	RHS	Support	Confidence	Lift
[1]	{}	=> {589}	0,096545436	0,09654544	1,00000000
[2]	{}	=> {80}	0,066659913	0,06665991	1,00000000
[3]	{}	=> {590}	0,058251444	0,05825144	1,00000000
[4]	{590}	=> {589}	0,002937899	0,05043478	0,5223943
[5]	{937}	=> {589}	0,002836592	0,12785388	1,3242872
[6]	{452}	=> {123}	0,002532671	0,14124294	4,6629065
[7]	{123}	=> {452}	0,002532671	0,08361204	4,6629065
[8]	{452}	=> {589}	0,002431365	0,13559322	1,4044498
[9]	{123}	=> {80}	0,002228751	0,07357860	1,1037908
[10]	{123}	=> {589}	0,002228751	0,07357860	0,7621137
[11]	{752}	=> {753}	0,001620910	0,48484848	87,0170799
[12]	{753}	=> {752}	0,001620910	0,29090909	87,0170799
[13]	{168}	=> {84}	0,001620910	0,18181818	47,2296651
[14]	{571}	=> {589}	0,001620910	0,06037736	0,6253777
[15]	{123}	=> {79}	0,001620910	0,05351171	1,3010198
[16]	{84}	=> {168}	0,001620910	0,42105263	47,2296651
[17]	{64}	=> {589}	0,001620910	0,14953271	1,5488325
[18]	{420}	=> {80}	0,001519603	0,12931034	1,9398517

Lampiran 18. *Output Top 23 Frequent Itemset Barang Umum*
 Algoritma Apriori untuk 1 Item

No	Items	Support
[1]	{256}	0.151858981
[2]	{72}	0.066659913
[3]	{71}	0.041130585
[4]	{557}	0.031303819
[5]	{39}	0.030594671
[6]	{1646}	0.030290751
[7]	{1255}	0.027555466
[8]	{1118}	0.026846317
[9]	{1299}	0.022186202
[10]	{1376}	0.020463985
[11]	{424}	0.020160065
[12]	{1057}	0.017931314
[13]	{1056}	0.015196029
[14]	{1597}	0.014689494
[15]	{393}	0.014182960
[16]	{1298}	0.013879040
[17]	{103}	0.013777733
[18]	{1649}	0.012562050
[19]	{466}	0.011852902
[20]	{1511}	0.011751596
[21]	{838}	0.011751596
[22]	{1651}	0.011751596
[23]	{110}	0.011447675

Lampiran 19. *Output Frequent Itemset Data Barang Umum*
Algoritma Apriori untuk 2 Item

No	Items	Support
<hr/>		
[1]	{1299,256}	0.003545740
[2]	{1057,256}	0.003140513
[3]	{1646,256}	0.003039206
[4]	{256,72}	0.002937899
[5]	{1255,256}	0.002836592
[6]	{256,71}	0.002735285
[7]	{256,424}	0.002532671
[8]	{1057,1646}	0.002532671
[9]	{103,256}	0.002228751
[10]	{1646,72}	0.002228751
[11]	{1118,256}	0.002127444
[12]	{224,256}	0.002026137
[13]	{112,256}	0.001722217
[14]	{1056,256}	0.001722217
[15]	{1255,72}	0.001722217
[16]	{1479,1480}	0.001620910
[17]	{636,637}	0.001620910
[18]	{1298,256}	0.001620910
[19]	{1646,71}	0.001620910
[20]	{1649,256}	0.001519603
[21]	{1651,72}	0.001519603
[22]	{256,39}	0.001519603
[23]	{71,72}	0.001519603

Lampiran 20. *Output Top 18 Rule Algoritma Apriori Data Barang Umum*

No	LHS	RHS	Support	Confidence	Lift
[1]	{}	=> {256}	0.151858981	0.15185898	1.0000000
[2]	{}	=> {72}	0.066659913	0.06665991	1.0000000
[3]	{1299}	=> {256}	0.003545740	0.15981735	1.0524063
[4]	{1057}	=> {256}	0.003140513	0.17514124	1.1533150
[5]	{1646}	=> {256}	0.003039206	0.10033445	0.6607080
[6]	{1255}	=> {256}	0.002836592	0.10294118	0.6778735
[7]	{71}	=> {256}	0.002735285	0.06650246	0.4379225
[8]	{424}	=> {256}	0.002532671	0.12562814	0.8272684
[9]	{1057}	=> {1646}	0.002532671	0.14124294	4.6629065
[10]	{1646}	=> {1057}	0.002532671	0.08361204	4.6629065
[11]	{103}	=> {256}	0.002228751	0.16176471	1.0652298
[12]	{1646}	=> {72}	0.002228751	0.07357860	1.1037908
[13]	{1118}	=> {256}	0.002127444	0.07924528	0.5218347
[14]	{224}	=> {256}	0.002026137	0.20833333	1.3718868
[15]	{1056}	=> {256}	0.001722217	0.11333333	0.7463064
[16]	{1255}	=> {72}	0.001722217	0.06250000	0.9375950
[17]	{112}	=> {256}	0.001722217	0.15887850	1.0462240
[18]	{1479}	=> {1480}	0.001620910	0.48484848	87.0170799

Lampiran 21. Output Top 18 Rule Algoritma Apriori Data Barang Umum

SURAT KETERANGAN

Saya yang bertanda tangan di bawah ini menerangkan bahwa :

1. Mahasiswa Statistika FMIPA-ITS dengan identitas berikut :

Nama : SITI QOMARIYAH

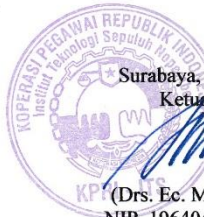
NRP : 1313100026

Telah mengambil data di instansi/perusahaan kami :

Nama Instansi : KPRI ITS

sejak tanggal 10 Oktober 2016 sampai dengan 28 Januari 2017 untuk keperluan Tugas Akhir/ Thesis Semester Gasal/Genap* 2016/ 2017.

2. Tidak Keberatan/Keberatan* nama perusahaan dicantumkan dalam Tugas Akhir/ Thesis mahasiswa Statistika yang akan di simpan di Perpustakaan ITS dan dibaca di lingkungan ITS.
3. Tidak Keberatan/Keberatan* bahwa hasil analisis data dari perusahaan dipublikasikan dalam E journal ITS yaitu Jurnal Sains dan Seni ITS.



*Surabaya, 23 Januari 2017

Ketua KPRI ITS

(Drs. Ec. Muiriyono, M.Si)
NIP. 19640408 198702 1 001

*(coret yang tidak perlu)

BIODATA PENULIS



Penulis dengan nama lengkap Siti Qomariyah dilahirkan pada tanggal 6 Juli 1995 di kota Pati. Penulis merupakan anak kedua dari dua bersaudara. Ayah penulis bernama Rifai dan Ibu penulis bernama Sulasmi. Penulis mulai menempuh pendidikan pertama yaitu di MI Matholiul Ulum pada tahun 2001, lalu melanjutkan di SMPN 3 PATI, dan kemudian melanjutkan di SMAN 1 Pati. Lalu pada tahun 2013 penulis diterima di program studi S1 jurusan Statistika ITS. Selama perkuliahan penulis juga mengikuti organisasi, turut serta dalam kepanitiaan dan mengikuti pelatihan ataupun seminar di ITS. Pada tahun pertama penulis mengikuti UKM IBC, pada tahun kedua penulis menjadi staff *public relation* di divisi SCC HIMASTA-ITS dan pada tahun ketiga penulis menjadi ketua divisi SCC HIMASTA-ITS. Selain mengikuti organisasi, penulis pernah menjadi asisten dosen mata kuliah komputasi statistik dan Sistem Informasi Managemen (SIM). Hobi penulis yaitu mendaki gunung, bermain badminton, *hunting* foto, dan menyanyi. Untuk dapat mengajak penulis berdiskusi mengenai tugas akhir ini, penulis dapat dihubungi pada alamat email sitidotkom@gmail.com atau melalui 085640617682.